FACULTY OF ENGINEERING AND ARCHITECTURE

Stretch measurement for online quality control in a web processing machine using low-cost vision

Daan Van Reepingen
Student number: 02103598

Supervisors: Prof. dr. ir. Jeroen De Kooning, Prof. dr. ir. Guillaume Crevecoeur
Counsellors: Yentl Thielemans, Jos Bruggeman

Master's dissertation submitted in order to obtain the academic degree of
Master of Science in Electrical Engineering Technology - Automation

Academic year 2024-2025

GHENT UNIVERSITY

Acknowledgment

From September 2024 to May 2025, I had the opportunity to conduct my master's thesis at Ghent University, spending most of my time at the vibrant Kortrijk campus. This thesis marks the end of my studies in Electrical Engineering – Automation and reflects both the technical challenges I faced and the many meaningful connections I made along the way.

Throughout this journey, I worked in a fantastic environment. The atmosphere in Kortrijk was not only professional and productive, but also warm and welcoming. The well-equipped labs and high-tech tools made it a joy to work on my project each day, and the team's enthusiasm elevated the entire experience.

I'm especially grateful to my counsellors, Jos Bruggeman and Yentl Thielemans, for their outstanding support and generosity. They consistently went above and beyond – staying late so I could complete tests, helping me on weekends while I wrote my thesis, and even assisting with other courses and tasks. Their advice on writing and presenting was invaluable, and their encouragement made a real difference throughout the year.

I also want to thank my academic supervisors, Prof. Dr. ir. Guillaume Crevecoeur and Prof. Dr. ir. Jeroen De Kooning, whose expertise and guidance helped shape the quality and direction of this work.

Special thanks go to Kwinten Mortier, Jasper De Viaene, Arul Mathivanan, and the rest of the Kortrijk team for their technical help, good conversations, and for always making me feel welcome. It was a privilege to be part of such a driven and friendly group.

I'm also thankful to my friends and colleagues – Wout De Vuyst, Brecht De Bleeckere, Robbe Dhondt, and Fabian De Petter – for the laughter, distractions, and support that made the long days easier. A special thank you goes to Jef Bruggeman, one of my best friends, not only for his constant support but also for being Jos's brother – otherwise I might never have heard about this thesis opportunity.

Lastly, I thank my family for their unwavering support, especially my parents for always believing in me, and my grandfather, whose pride in my studies was a constant motivation. Knowing how much it meant to him to see me graduate kept me going until the end.

To everyone who contributed in ways big or small – thank you.

Explanation regarding the master's thesis and the oral presentation

This master's dissertation is part of an exam. Any comments formulated by the assessment committee during the oral presentation of the master's dissertation are not included in this text.

Permission for consultation

The author gives permission to make this master dissertation available for consultation and to copy parts of this master dissertation for personal use. In all cases of other use, the copyright terms have to be respected, in particular with regard to the obligation to state explicitly the source when quoting results from this master dissertation.

May 22, 2025

Transparency Statement

During the preparation of this work the author utilised ChatGPT 4o and Elicit.org to find and summarise sources. After using this service, the author reviewed and edited the content as needed and takes full responsibility for the content of the publication.

Abstract

This thesis investigates the potential of low-cost, vision-based technology for use in web processing machines. Specifically, it examines whether such systems can offer a reliable and precise alternative to traditional stretch sensors. The central aim is to develop a system capable of real-time deformation monitoring that is affordable, easy to integrate, and sufficiently precise for practical use in industrial settings. This goal responds to the growing demand for flexible and cost-effective quality control solutions within the broader context of smart manufacturing.

The proposed sensor is evaluated in terms of its precision, responsiveness, and robustness across a range of conditions and material types. It is benchmarked against a commercial ultrasonic edge sensor to assess comparative performance. Results indicate that the vision-based approach achieves micrometre-level precision, with significantly reduced noise and improved signal stability. Moreover, its integration into a closed-loop control system demonstrates potential for real-time tension adjustment, contributing to improved material uniformity during production.

Some limitations remain, particularly in terms of processing latency and synchronisation. Nonetheless, this work provides evidence that low-cost embedded vision systems can support quality monitoring and control in web processing applications. The findings suggest that such systems may serve as a viable alternative in environments where conventional sensors are either impractical or prohibitively expensive.

Extended Abstract

Daan Van Reepingen daan.vanreepingen@ugent.be, Yentl Thielemans *Member, IEEE,*, Jos Bruggeman *Member, IEEE,*, Guillaume Crevecoeur, *Senior Member, IEEE*, and Jeroen D. M. De Kooning, *Senior Member, IEEE,*

***Abstract*—High-precision, non-contact stretch estimation in web processing machines (WPMs) is often limited to expensive, specialised systems. This research presents the development of a low-cost, vision-based sensor for real-time transverse stretch measurement in WPMs. The system combines gradient-based edge detection with Frequency Domain Zero Padding (FDZP) to achieve subpixel precision using Raspberry Pi (RPi) hardware. While designed for affordability and ease of integration, the sensor demonstrates micrometre-level precision under controlled conditions. Comparative experiments indicate that the proposed approach may offer improved noise robustness and signal quality relative to a commercial ultrasonic edge sensor. Final integration into a closed-loop tension control system suggests that the method holds potential for scalable implementation in cost-sensitive industrial contexts.**

***Keywords*—web processing machine, vision sensor, stretch measurement, subpixel edge detection, closed-loop control, Industry 4.0**

I. Introduction

High-precision, non-contact stretch estimation in WPMs is typically limited to expensive, specialised systems. These systems can be impractical in cost-sensitive industrial settings, particularly when real-time feedback or closed-loop control is required. This work aims to address the lack of affordable solutions in stretch sensing. It investigates a low-cost, vision-based alternative that may offer comparable precision while remaining suitable for embedded deployment.

This challenge emerges in the broader context of Industry 4.0, which encourages the transition toward adaptable, efficient, and traceable production systems. Cyber-physical systems and smart manufacturing play a key role in enabling this transition through enhanced sensing, data acquisition, and feedback capabilities [1] [2]. In WPMs, precise stretch measurement can help maintain product quality, reduce waste, and support sustainable manufacturing practices [3]. However, real-time measurement of material deformation remains technically challenging, particularly when relying on embedded or low-cost hardware.

Many existing stretch sensing methods, such as strain gauges [4], require direct contact with the material. This physical interface complicates continuous or localised measurement and is difficult to maintain within the moving environment of a WPM. In such applications, non-contact techniques are typically preferred. Among these, digital image correlation (DIC) is utilised in research and high-end industrial contexts for estimating stretch based on surface tracking. However, DIC systems are generally costly, complex, and unsuitable for real-time integration on embedded platforms due to their reliance on high-speed imaging and proprietary software [5].

This work proposes a low-cost alternative that aims to replicate partial performance characteristics of DIC without its financial limitations. The system targets a precision between 20 and 100 µm/m and operates at frame rates between 50 and 500 Hz [6] [7], utilising RPi modules with integrated cameras. Transverse stretch is estimated by detecting material edges and refining their position through FDZP, enabling subpixel precision with lightweight, embedded processing. This approach may offer a scalable, robust, and more accessible solution for continuous stretch monitoring inside the WPM.

Experimental benchmarking compares the proposed system with a commercial ultrasonic edge sensor. Results suggest that the vision-based method performs favourably in terms of responsiveness and signal quality. Final deployment in a real WPM environment demonstrates its robustness and highlights its potential for use in closed-loop control applications. This work supports the broader adoption of embedded vision for quality monitoring in industrial settings where cost, adaptability, and ease of integration are critical.

II. Vision-based stretch sensor development

Material stretch is estimated by detecting the lateral position of its edges using two camera modules mounted on opposite sides of a roller inside the WPM, as seen in Fig. 1. Each camera performs real-time edge detection on the visible edge of the material. By subtracting the two edge positions from the known roller length, the system calculates the current material width. Changes in this width over time reflect transverse stretch. To extract edge positions using low-cost vision hardware, each image is reduced to a single horizontal pixel line at the middle of the roller. This line is converted to greyscale and analysed using the Luma channel to compute the gradient [8]. Peaks in this gradient correspond to material edges, as shown in Fig. 2. Using this course edge detection a symmetrical region of interest (ROI) is extracted and transformed using a Fast Fourier Transform (FFT) [9]. The frequency resolution of this discrete signal is inherently limited by the Nyquist frequency [10], which in this case is correlated to the pixelsize. By zero-padding the spectrum, appending zeros beyond the original frequency bins, the total length increases. This effectively simulates a higher sampling rate for the same signal

Daan Van Reepingen, Yentl Thielemans, Jos Bruggeman, Guillaume Crevecoeur and Jeroen D. M. De Kooning are with the Dynamical Systems & Control group (DySC), Department of Electromechanical, Systems & Metal Engineering, Ghent University, Tech Lane Ghent Science Park - Campus A, Technologiepark Zwijnaarde 131, B-9052 Ghent, Belgium and with the FlandersMake@UGent.be - Corelab Miro, Flanders Make, Belgium. e-mail: daan.vanreepingen@ugent.be

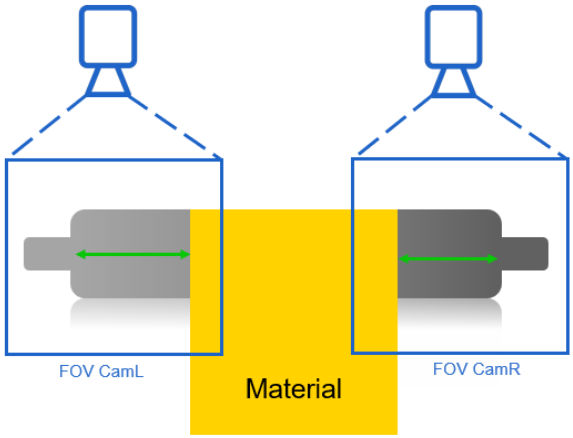

Fig. 1. Camera setup with 2 RPi modules each measuring one of the material edges on the roller

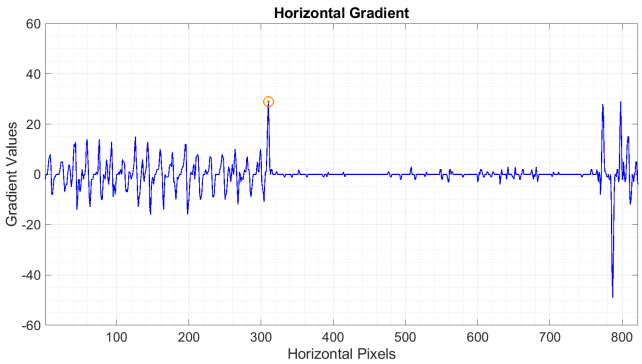

Fig. 2. Pixelline gradient showing transition from material to roller and resulting edge peak value (highlighted)

shape. This refined frequency representation enables denser resampling of the signal when applying the inverse FFT. This results in a higher-resolution time-domain signal from which the edge position can be estimated more precisely. The gain in resolution is governed by the subpixel Factor shown in eq. 1

$$\text{Factor} = \frac{\#\ \text{subpixels}}{\#\ \text{pixels}} \tag{1}$$

This technique is called Frequency Domain Zero Padding (FDZP) [11]. While FDZP introduces no new information, it yields precise interpolation well-suited for one-dimensional signals. Higher factors improve precision but increase computational load. This is a key trade-off on embedded hardware like the RPi version 4B. Despite this constraint, FDZP is central in achieving the subpixel precision demonstrated throughout this work. This combination of basic edge detection and FDZP enables subpixel precision without the need for expensive hardware.

The full program is split into a preprocessing and processing stage. During preprocessing, high-resolution images are used to calibrate key parameters: the middle of the roller, a pixel-to-millimetre conversion factor, and the horizontal location of the roller's physical end. This pixel-to-mm conversion is essential, as all edge measurements are initially made in pixel space. Several strategies were evaluated. Measuring the apparent roller width was found to be unreliable due to tangential distortion effects, previously calculated in [12]. Instead, the most robust approach used a marker with a known dimension which was applied to the roller. This conversion was further improved by applying intrinsic camera calibration to compensate for radial and perspective distortions [13] [14].

Once calibration is complete, the real-time processing loop captures low-resolution images and performs continuous edge detection. Key performance gains were achieved using parallelisation techniques. Tasks such as image capture, FDZP processing, and data transmission were distributed across the RPi's CPU cores using a dynamic worker pool [15] [16]. Additional improvements were made by implementing a multithreaded architecture. This separates image acquisition and processing utilising a producer-consumer model as illustrated in Fig. 3. Together, these methods improved the cycle time to 30 Hz, the hardware limit of the RPi version 4B.

Fig. 3. Schematic showing process flow with grouped multithreaded tasks

Selecting an appropriate FDZP scaling factor is key to balancing precision and responsiveness. As shown in Fig. 4, increasing the Factor improves theoretical precision but also raises computational demands. Beyond a certain point, the gain in precision is limited by factors such as edge roughness and imperfect focus, which degrade the effective signal-to-noise ratio. A FDZP factor of 409.6 was selected as the optimal operating point, offering subpixel precision without excessive slowdown. Precision also varies with material type; thin paper and mesh produced sharper edges and lower deviation than textured surfaces like waterproofing membrane.

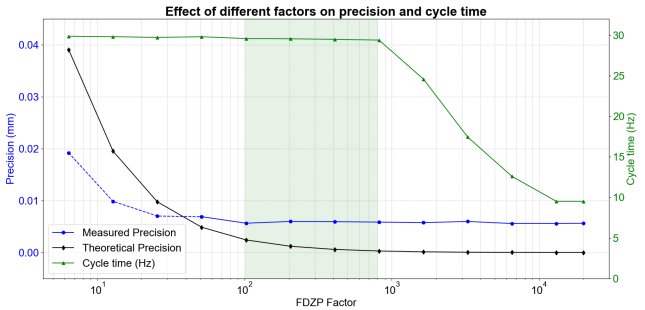

Fig. 4. FDZP Factor working point and measured cycle time and precision

The system was benchmarked under steady-state conditions following ISO 5725-1:2023 [17]. Both cameras were evaluated individually, then jointly as a stretch sensor. The cameras

achieved a standard deviation of 3.98 µm and 7.42 µm, while the ultrasonic edge sensor [18] measured 22.35 µm. Results are summarised in Table I. The ultrasonic sensor also exhibited drift and failed to detect thin, semi-transparent materials. In contrast, the vision-based sensor remained reliable across all tested materials, reinforcing its applicability in diverse industrial conditions.

TABLE I. SUMMARY OF MEASUREMENT STATISTICS FOR EACH SENSOR

Sensor	SD (µm)	Max (µm)	Min (µm)	Range (µm)
CamL	3.978	13.547	-13.186	26.733
CamR	7.417	39.497	-34.729	74.226
Ultrasonic	22.349	154.074	-120.926	275.000

Finally, the width precision was calculated by combining the measurements of both cameras. The theoretical value of 8.06 µm, based on standard deviation propagation, was closely matched by the measured precision of 8.43 µm .

III. PRACTICAL SENSOR EVALUATION

To evaluate the RPi stretch sensor under realistic industrial conditions, a set of comparative and characterisation tests was conducted using moving material inside the WPM. The first stage involved comparing the sensor's output with that of a dual ultrasonic edge detection setup [18]. Both sensors measured material width simultaneously during continuous motion at 10 m/min under 80 N tension. The measurement results are shown in Figs. 5 and 6. A Pearson correlation coefficient [19] of $r = 0.8319$ indicated a strong linear relationship between the two signals. This suggests that both sensors captured the same global deformation trends.

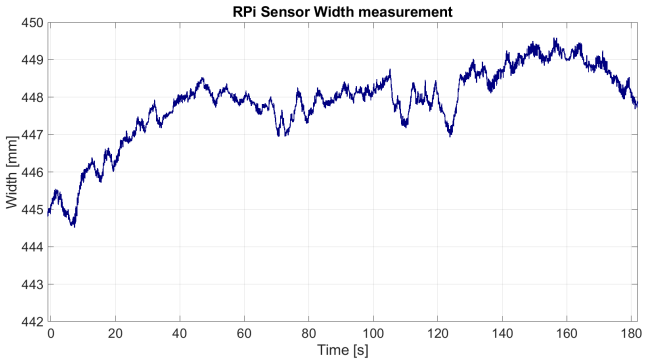

Fig. 5. RPi sensor width measurement on moving material

A detailed frequency-domain analysis was conducted to assess signal clarity and sensor stability. Although both sensors reported an identical standard deviation of 1.02 mm, this metric proved insufficient to distinguish between noise and meaningful variation. The Signal-to-Noise Ratio (SNR) offered more insight: the RPi sensor achieved a value of 14.74, nearly three times higher than the ultrasonic sensor's 5.47, reflecting its cleaner output. The high-frequency energy, defined as spectral energy above 2 Hz, was also markedly lower for the RPi

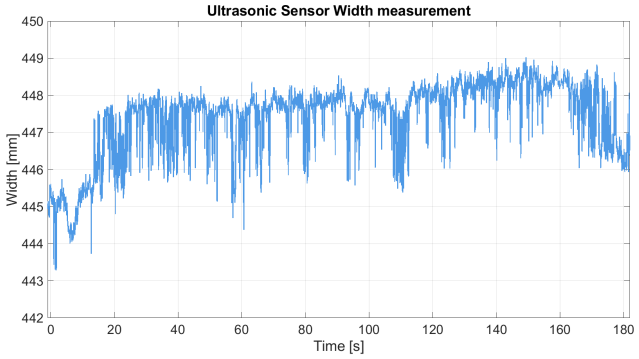

Fig. 6. Ultrasonic sensor width measurement on moving material

sensor (19,411 vs. 99,119). This confirms a reduction in sensor jitter and better noise suppression. Lastly, the noise standard deviation, calculated from high-pass residuals, was 0.075 mm for the RPi sensor versus 0.412 mm for the ultrasonic sensor, over five times lower. These results, summarised in Table. II, position the RPi sensor as more precise, stable, and robust under motion.

TABLE II. COMPARISON OF SIGNAL QUALITY METRICS BETWEEN ULTRASONIC AND RPI SENSORS

Metric	Ultrasonic Sensor	RPi Sensor
SD [mm]	1.02	1.02
SNR [–]	5.47	14.74
High-Frequency Energy [a.u.]	99,119	19,411
Noise SD [mm]	0.412	0.075

To further validate its dynamic performance, the RPi sensor was used to construct a force–extension curve for kerdi membrane material [20] during web processing. The membrane was subjected to gradually increasing tension. The stretch was measured in real-time and residual deformation recorded at a reference tension of 20 N, both shown in Fig. 7. This method separated elastic and plastic components of deformation. The data revealed a near-linear relationship between applied force and transverse stretch, with approximately 1.0 mm extension observed at 200 N over a 450 mm width, shown in Fig. 8. No discontinuities or abrupt slope changes were detected, consistent with the ductile behaviour of polyethylene-based laminates [21]. Residual stretch after unloading confirmed gradual plastic deformation, demonstrating the sensor's sensitivity and coherence under dynamic conditions. While some minor non-linearity arose due to repeated use of the same material segment, the results still provide a valid proof of concept. The system accurately captured both short-term elasticity and long-term plastic response. This reinforces the suitability of the RPi sensor for industrial web-processing applications.

IV. QUALITY CONTROL INSIDE THE WPM

The final phase of this study integrates the RPi stretch sensor into a closed-loop feedback control system within the WPM. The objective is to dynamically adjust the applied tension

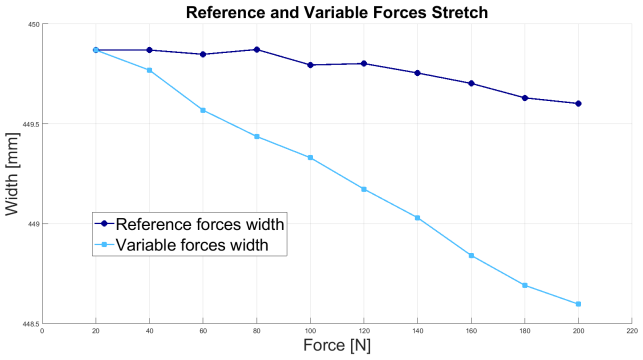

Fig. 7. Average measured width during force application and after force removal

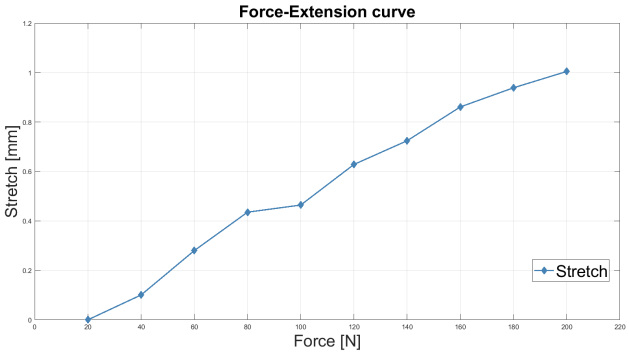

Fig. 8. Resulting force-extension curve from measurement results

based on real-time width measurements. This with the goal of improving material quality by reducing large-scale deformation during processing.

A proportional controller is placed upstream of the WPM's existing PI-controlled dancer system, forming a cascaded architecture. The RPi sensor measures the material width, which is filtered and compared to a setpoint. The resulting error adjusts the tension setpoint of the dancer unit. This approach replaces the static base tension with a responsive value, enabling the system to correct for slow, undesirable stretch patterns. However, a delay of approximately 3 seconds between the dancer and the measurement point limits the controller's responsiveness, effectively restricting corrections to low-frequency deformations. The gain for this controller is estimated using the force–extension curve seen in Fig. 8. Here a stiffness of approximately 175 N/mm was calculated. Initial tests used a gain of 100, later reduced to 80, both to account for material degradation. This method avoids trial-and-error tuning and strengthens the link between characterisation and control. Figure 9 shows a comparison in material width with and without control and their respective tension setpoints.

Frequency-domain analysis offers the clearest insight into how the controller affects material deformation. A Fourier transform was applied to the measured width signal. Its spectral energy was divided into three bands. Figure 10 shows their values. In the low-frequency band (below 0.20 Hz), which

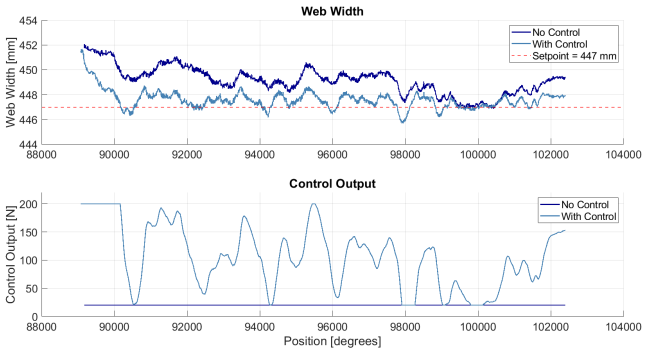

Fig. 9. Material width in function of position inside the WPM with and without tension control towards width-setpoint and their respective tension setpoints

represents large-scale warps, the spectral energy decreased significantly from 0.595 to 0.274 a.u.. This confirms that the control system effectively suppresses broad, slow deformations in the material. In the control-frequency band (0.20–0.40 Hz), the energy increased from 0.0067 to 0.0158 a.u.. This rise is expected and reflects two complementary effects: small overshoots or ripple caused by the controller operating near its bandwidth limit, and the finite reaction time of the system, which shifts some of the low-frequency energy into this band as the controller responds to deformations. Rather than being suppressed instantly, deviations are gradually corrected. This correction occurs through compensatory tension changes that manifest at slightly higher frequencies. High-frequency energy (above 0.40 Hz) increased only marginally (0.0091 to 0.0126 a.u.). This is consistent with the fact that the controller does not actively suppress these components and may introduce minor actuation-related dynamics. Overall, the total spectral energy was halved from 0.611 to 0.302 a.u.. This indicates that the system meaningfully reduces deformation energy without introducing excess noise.

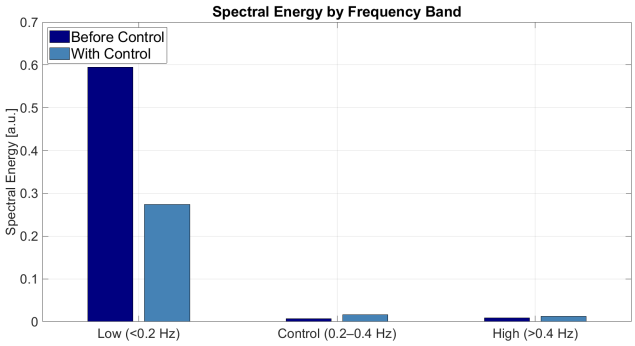

Fig. 10. Arbitrary deformation value for comparing material quality with and without tension control

Finally, some nuance is required in interpreting the results. While straightening the material is used as a quality proxy, excessive tensioning could lead to plastic deformation. The

goal is not to flatten the material at all costs but to compensate for machine-induced imperfections. However, the WPM used for testing did not contain any functional components that could cause such deformations. As a result, no specific tests were conducted to compensate for machine-induced tensions. Despite this, the control loop successfully demonstrated its ability to respond to real-time feedback under challenging conditions.

V. Conclusion

This work demonstrates that low-cost, vision-based transverse stretch sensing on embedded hardware can achieve micrometre-level precision. Utilising RPi cameras and FDZP-based subpixel edge detection, the system achieved a measured precision of 8.43 µm, closely matching theoretical expectations.

Compared to a commercial ultrasonic sensor, the proposed method showed improved signal quality, reduced noise, and higher robustness across material types. Frequency-domain analysis revealed a threefold increase in signal-to-noise ratio and a fivefold reduction in high-frequency noise, supporting its use as a practical, non-contact alternative for stretch measurement in WPMs.

The sensor was also integrated into a closed-loop control system, demonstrating the ability to suppress slow, large-scale deformations. Although limited by system delay, the controller meaningfully reduced low-frequency deformation energy without introducing excess noise.

Overall, the system offers a viable solution for real-time quality monitoring in cost-sensitive industrial settings. It aligns with Industry 4.0 goals by enabling affordable, adaptable, and embedded sensing. Some limitations remain, such as latency and processing speed. Nevertheless, the results establish a strong proof of concept and provide a foundation for further development.

References

[1] N. Jazdi, "Cyber physical systems in the context of industry 4.0," in *2014 IEEE International Conference on Automation, Quality and Testing, Robotics*. IEEE, 2014, pp. 1–4.

[2] H. Kagermann, W. Wahlster, and J. Helbig, "Securing the future of german manufacturing industry: Recommendations for implementing the strategic initiative industrie 4.0," acatech – National Academy of Science and Engineering, Frankfurt/Main, Germany, Tech. Rep., April 2013, final report of the Industrie 4.0 Working Group. [Online]. Available: https://www.acatech.de/publikation/securing-the-future-of-german-manufacturing-industry/

[3] A. Papavasileiou, G. Michalos, and S. Makris, "Quality control in manufacturing – review and challenges on robotic applications," *International Journal of Computer Integrated Manufacturing*, vol. 38, no. 1, pp. 79–115, 2025. [Online]. Available: https://doi.org/10.1080/0951192X.2024.2314789

[4] eFunda, "Strain gage sensitivity," n.d. [Online]. Available: https://www.efunda.com/designstandards/sensors/strain_gages/strain_gage_sensitivity.cfm

[5] H. Xiufeng, W. Jihang, and Z. Zhenhua, "Research on application of digital image correlation method," in *2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, 2020, pp. 838–840.

[6] J. Carr, J. Baqersad, C. Niezrecki, and P. Avitabile, "Predicting dynamic strain on wind turbine blade using digital image correlation techniques in conjunction with analytical expansion methodologies," *Proceedings of the 31st International Modal Analysis Conference (IMAC)*, 2013. [Online]. Available: https://www.researchgate.net/publication/292783598_Predicting_Dynamic_Strain_on_Wind_Turbine_Blade_Using_Digital_Image_Correlation_Techniques_in_Conjunction_with_Analytical_Expansion_Methodologies

[7] R. E, A. A, V. V, G. T, S. S, and S. Prabha R, "Strain measurement using digital image correlation technique," in *2023 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, 2023, pp. 1–8.

[8] International Telecommunication Union, "Recommendation ITU-R BT.601-7: Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios," International Telecommunication Union, Tech. Rep. BT.601-7, 2011. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/bt/r-rec-bt.601-7-201103-i!!pdf-e.pdf

[9] M. Heideman, D. Johnson, and C. Burrus, "Gauss and the history of the fast fourier transform," *IEEE ASSP Magazine*, vol. 1, no. 4, pp. 14–21, 1984.

[10] L. Auslander and F. A. Grunbaum, "The fourier transform and the discrete fourier transform," *Inverse Problems*, vol. 5, pp. 149–164, 1989. [Online]. Available: https://api.semanticscholar.org/CorpusID:121312591

[11] Y. Thielemans, N. Kayedpour, A. Coene, G. Crevecoeur, and J. D. M. De Kooning, "Speed and position measurement of rotating machinery with a triangular pattern vision encoder," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1–13, 2025.

[12] J. Bruggeman, "Operational motion monitoring of web processing machines using low-cost vision for adaptive control," 2024, master of Science in Electromechanical Engineering. [Online]. Available: http://lib.ugent.be/catalog/rug01:003214547

[13] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[14] O. Team, "Camera calibration with opencv," 2024. [Online]. Available: https://docs.opencv.org/4.x/dc/dbb/tutorial_py_calibration.html

[15] Python Software Foundation, "multiprocessing — process-based parallelism," n.d. [Online]. Available: https://docs.python.org/3/library/multiprocessing.html

[16] F. Doglio, *Parallel Programming with Python*, 2nd ed. Packt Publishing, 2014. [Online]. Available: https://www.packtpub.com/product/parallel-programming-with-python-9781783288397

[17] *Accuracy (trueness and precision) of measurement methods and results — Part 1: General principles and definitions*, International Organization for Standardization Std. ISO 5725-1:2023, July 2023. [Online]. Available: https://cdn.standards.iteh.ai/samples/69418/a7c3fb78cdce4a6398bd3f759670d72e/ISO-5725-1-2023.pdf

[18] M. GmbH, "bks+6/fiu ultrasonic web edge sensor," https://sensorpartners.com/en/product/microsonic-bks6-fiu/, 2025.

[19] Statistics How To. (2024) Correlation coefficient: Simple definition, formula, easy steps. [Online]. Available: https://www.statisticshowto.com/probability-and-statistics/correlation-coefficient-formula/

[20] Schluter Systems, "Schluter®-KERDI: Waterproofing Membrane," https://www.schluter.com/schluter-us/en_US/Membranes/Waterproofing-%28KERDI%29/Schluter%C2%AE-KERDI/p/KERDI, 2025.

[21] (2024) Polyethylene (pe) — properties, structures, uses, & facts. Encyclopædia Britannica. [Online]. Available: https://www.britannica.com/science/polyethylene

Daan Van Reepingen was born in Zottegem, Belgium in 2003. He is a thesis student at the department of Electromechanical, Systems & Metal Engineering of Ghent University. He has work-related experiences in factory-wide MES systems and beamsaw layout optimisation software at Ardis NV. His research interests include Vision techniques, PLC-driven machines, Robotics, Embedded systems engineering and additive manufacturing practices.

Yentl Thielemans was born in Kortrijk, Belgium in 1999. He received the Master degree in Electrical Engineering Technology from Ghent University, Belgium, in 2022. He is Ph.D. student at the department of Electromechanical, Systems & Metal Engineering of Ghent University, and affiliated with the FlandersMake@UGent MIRO core lab. His research interests include Vision techniques, Machine learning, Greybox models, Digital Twins and transforming industrial machines regarding Industry 4.0.

Jos Bruggeman was born in Aalst, Belgium in 2000. He received the Master degree in Electrical Engineering Technology from Ghent University, Belgium, in 2022. He is Ph.D. student at the department of Electromechanical, Systems & Metal Engineering of Ghent University, and affiliated with the FlandersMake@UGent MIRO core lab. His research interests include Vision techniques, PLC-driven machines, Automation and transforming industrial machines regarding Industry 4.0.

Guillaume Crevecoeur (°1981) received his Master and PhD degree in Engineering Physics from Ghent University in 2004 and 2009, respectively. He received a Research Foundation Flanders postdoctoral fellowship in 2009 and was appointed Associate Professor at Ghent University in 2014. He is member of Flanders Make in which he leads the Ghent University activities on machines, intelligence, robotics and control. With his team, he conducts research on system identification and nonlinear control for mechatronics, industrial robotics and energy systems. His goal is to endow physical dynamic systems with improved functionalities and capabilities when interacting with uncertain environments, other systems and humans.

Jeroen D. M. De Kooning (M'09, SM'20) was born in Kapellen, Belgium, in 1987. He received the Bachelor, Master and Ph.D. degrees in electromechanical engineering from Ghent University, Belgium, in 2008, 2010 and 2015 respectively. Since 2019, he is assistant professor at the Department of Electromechanical, Systems and Metal Engineering at the Faculty of Engineering and Architecture of Ghent University, Belgium. In 2022, he was a visiting professor at the Lappeenranta University of Technology, Finland. He is a member of the Dynamical Systems and Control (DySC) research group and the FlandersMake@UGent MIRO core lab. He conducted research on current waveform shaping techniques for permanent magnet synchronous machines, and optimal control and design of renewable energy systems. His present research interests include modelling, optimisation and control of mechatronic systems, drivetrains and manufacturing machines in an Industry 4.0 context, with a particular interest in digital twins.

Contents

List of Figures

List of Tables

List of Acronyms

a.u. Arbitrary Unit.

CamL Camera Left.

CamR Camera Right.

CPS cyber-physical systems.

CPU Central Processing Unit.

CWD Controllable Width Deviations.

DIC Digital Image Correlation.

FDZP Frequency Domain Zero Padding.

FFT Fast Fourier Transform.

FOV Field Of View.

GPU Graphics processing unit.

I/O Input/Output.

IFFT Inverse Fast Fourier Transform.

IP Internet Protocol.

ISO International Organization for Standardization.

OPC UA Open Platform Communications Unified Architecture.

PE Polyethylene.

PLC Programmable Logic Controller.

RGB Red Green Blue.

RMSE Root Mean Square Error.

ROI Region of Interest.

RPi Raspberry Pi.

RTC Real Time Clock.

SD Standard Deviation.

SDG Sustainable Development Goal.

SNR Signal to Noise Ratio.

TCP Transmission Control Protocol.

WPM Web Processing Machine.

1

Introduction

The increasing demand for higher efficiency, traceability, and adaptability in industrial production drives a rapid transformation in manufacturing technologies under the umbrella of Industry 4.0. This evolution centres on the integration of cyber-physical systems (CPS) [1] with advanced sensing and control capabilities. These systems enable real-time decision-making, improve product quality, and optimise resource usage. Within this framework, vision-based quality control emerges as a promising non-invasive technique for monitoring and measuring dynamic processes in manufacturing environments.

Web Processing Machines (WPM), widely used in the production of textiles, plastics, and composites, require the ability to monitor and control material stretch in real time. This capability is considered crucial for maintaining product quality and process stability. Excessive or uneven stretching during production can cause deformation, misalignment, or functional defects. These problems may result in material waste and increased downtime. Accurate stretch measurement is therefore important not only for production efficiency but also as a potential contributor to sustainable manufacturing practices [2]. Reducing defects and waste at the source helps conserve energy, minimise raw material usage, and reduce rework or scrap. These effects align with broader sustainability goals in modern industry.

This thesis addresses these challenges by developing a low-cost, camera-based system for measuring material stretch in web processing applications. The proposed solution uses embedded vision and lightweight image processing to perform real-time measurement with competitive precision. It targets applications where affordability, flexibility, and ease of integration are key considerations. The core idea is to combine accessible embedded cameras with lightweight processing techniques to estimate material deformation in real time. The overarching goal is to create a sensor system that op-

erates autonomously, delivers acceptable levels of precision, and integrates smoothly into industrial environments. The solution does not aim to replace high-end metrology systems. Instead, it suits scenarios where cost constraints, installation simplicity, or system adaptability take precedence over ultimate precision. By reducing hardware complexity and favouring software-based approaches, the system offers a promising route for extending quality control capabilities. This is especially valuable in sectors or use cases where traditional sensors are impractical or cost-prohibitive. In doing so, it contributes both to the evolution of measurement technologies and to the broader ambition of data-driven, adaptive manufacturing, a concept at the core of Industry 4.0 [3].

In addition to developing the sensing system, this thesis benchmarks the proposed approach against a commercially available ultrasonic edge sensor commonly used in industrial practice. The experimental evaluation compares both systems in terms of precision, responsiveness, and signal quality under static and dynamic conditions. This comparison provides a grounded, preliminary assessment of the industrial feasibility of the proposed vision-based method.

Finally, the sensor is deployed in a real-world use case to examine its potential role in closed-loop control or automated inspection within the WPM. By evaluating its behaviour under varying operational conditions, this work assesses the robustness and generalisability of the proposed sensor. The results contribute to the growing body of literature that supports the integration of affordable vision systems into smart manufacturing environments, and suggest that low-cost stretch sensing could become a valuable component in future quality control frameworks, potentially bridging the gap between high-precision measurement and economic feasibility.

2

Vision-based stretch sensor development

Measuring the stretch of a material within a WPM plays a critical role in maintaining product quality and optimising process efficiency. However, accurate and reliable stretch measurement in such an environment requires addressing several key considerations. Common stretch measuring sensors, such as strain gauges [4][5] or fabric-based textile stretch sensors [6], typically rely on physical contact with the material under tension. This reliance makes it difficult to perform continuous, localised measurements within the machine. Such measurements are essential for effectively monitoring and controlling material stretch throughout the manufacturing process. Implementing these sensors on the material inside the WPM proves challenging due to the electrical wiring they require. For moving materials, the only efficient way to measure stretch is through a non-contact approach. Many methods can achieve this, such as Digital Image Correlation (DIC). Another important consideration in stretch measurement is the type of stretch that needs to be measured. Inside the WPM, which typically processes textiles and other sheet-like materials, two directions of stretch are most prevalent: longitudinal stretch and transverse stretch. Many stretch or strain sensors, including the aforementioned strain gauge, cannot measure these two types of stretch completely independently. This limitation often arises from effects like Poisson's effect and material anisotropy [7][8]. When designing a stretch sensor, it is important to determine which type of stretch to measure and to consider whether coupling between stretch directions will pose a problem.

2.1 Industrial methods for stretch measurement

Digital Image Correlation is widely used in materials science and engineering because it provides full-field strain measurements with high accuracy, even for complex geometries and heterogeneous materials.

2 Vision-based stretch sensor development

In DIC, a high-resolution camera captures images of a material's surface before and during deformation. The surface can be speckled with a random pattern to serve as tracking points. The DIC software compares these images by analysing pixel displacements in small subsets (correlation windows) of the pattern. By tracking how these subsets move between images, the software calculates the displacement field. From this data, it determines the strain (including stretch) by measuring how much the distances between points on the material change over time [9].

The main disadvantage of DIC lies in the hardware and installation cost. Because the material is moving, cameras must operate at a sufficiently high frame rate to measure stretch efficiently. This requirement, combined with the licence cost of DIC software, results in a high financial entry point for using DIC inside the Web Processing Machine.

The objective for the stretch sensor developed in this thesis is to achieve performance comparable to a standard DIC setup. It targets a sensitivity range between 20 and 100 µm/m and a frame rate of 50 to 500 Hz [10] [11]. Attaining this level of precision requires a high spatial resolution of at least 4 megapixels. In addition to high resolution, the system must also deliver a high frame rate and use a global shutter. However, cameras that meet these specifications typically fall into the high-end price range. For instance, the Phantom v642, a high-speed camera with these capabilities, costs approximately 20,000 USD [12].

2.2 Low-cost vision approach to transverse stretch

The utilisation of low-cost vision hardware imposes certain constraints when developing a stretch sensor. The most significant limitations pertain to the processing power and frame rate of the camera. Lower processing power limits the resolution of individual images, and a lower frame rate prevents the creation of a DIC algorithm for the material speeds common in the WPM. This situation necessitates the implementation of an alternative approach.

An alternative method of measuring stretch is to measure the width of the material. When a force is applied in the longitudinal direction, the material will often elongate. As a result, in the transverse direction the material will often constrict. One can determine this constriction by measuring the material's width. The width of the material can be measured by applying an edge detection algorithm. The use of such algorithms on low-cost vision hardware has been previously examined in [13].

The main limitation of this approach is also one of its biggest advantages. It only allows measurement of stretch in the transverse direction, but conversely, it separates this from stretch in the longitudinal direction. This separation facilitates decoupling the different types of stretch.

2.2.1 Hardware placement inside the WPM

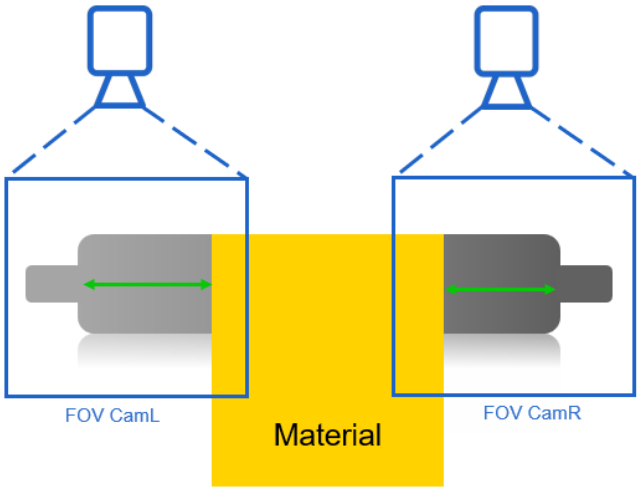

Figure 2.1: Camera setup with 2 RPi modules each measuring one of the material edges on the roller

To develop a stretch sensor for the material in the WPM, the first step is to determine the optimal placement of the hardware inside the machine. The two Raspberry Pi (RPi) version 4B units with camera modules are installed on the frame of the WPM using ISB aluminium profiles. Their placements is shown in Fig. 2.1

In this case, placing the stretch sensor at the roller proves most practical, as the roller provides a fixed reference point for edge measurements. The goal is to determine the width of the material by measuring the distances from the roller's edge to the material's edge on both sides. Subtracting these values from the total roller length yields the material's width. Beyond calculating width, this method also offers the added benefit of

determining the material's lateral position on the roller.

2.3 Edge detection methodology

Edge detection refers to the process of identifying boundaries within images by analysing variations in intensity values [14]. With the availability of modern Python libraries, engineers have access to powerful tools for implementing real-time edge detection systems. A wide range of methods exists for this purpose, including algorithmic approaches such as the Canny Edge Detector [15] [16], as well as techniques that involve image smoothing with Gaussian filters followed by gradient computation using Sobel operators [17].

This section provides a detailed explanation of the edge detection method developed for this thesis. Edge detection is used in various contexts within the stretch sensor. Any references to edge detection in the following chapters refer specifically to this method.

2.3.1 Extracting edges from greyscale pixellines

A standard image consists of five dimensions: the spatial dimensions of height and width in pixels, along with three colour channels: Red, Green, and Blue (RGB). In essence, an image comprises three matrices, one for each colour channel, each defined by height and width. To effectively plot the edge to be detected, these five dimensions can be reduced to two. This dimensionality reduction not only simplifies the problem but also significantly decreases the computational load required for edge detection.

The spatial dimensions can be simplified by extracting a single pixel line from the image. Selecting a horizontal or vertical line determines the direction of the edge that will be detected. However, this approach is less suited for detecting complex, angled edges. In such cases, and when the angle is known, Difference of Gaussians (DoG) filters [18] or Hough transformations provide a more effective solution. These techniques, however, will not be used in the stretch sensor.

To reduce the colour dimensions, the image is converted to greyscale. Various greyscale conversion methods were evaluated [19], but the luma conversion method provided by OpenCV, based on the ITU-R Recommendation BT.601-7 [20], proved to be the most effective for this application.

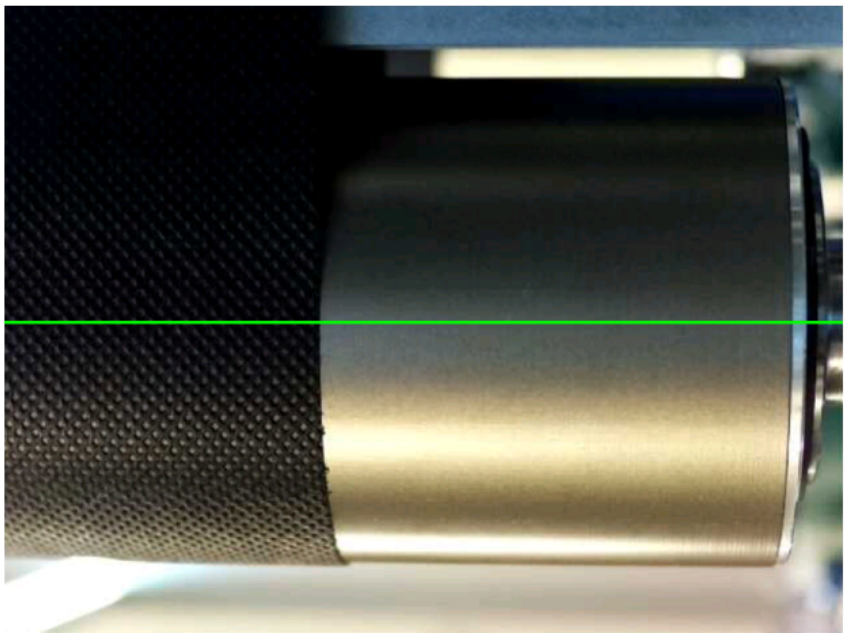

Figure 2.2: Roller with highlighted pixelline

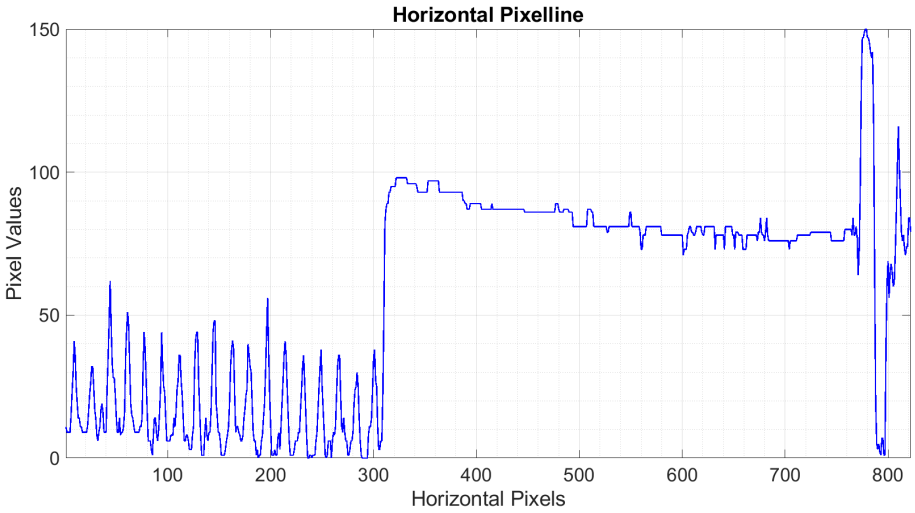

Figure 2.3: Horizontal pixelline plot with Luma pixelvalues showing clear boundry between material and roller

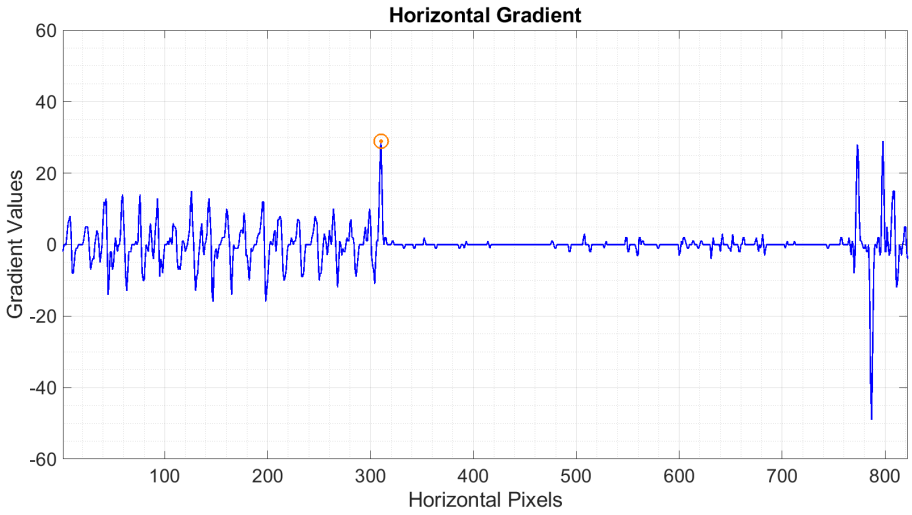

Figure 2.4: Gradient plot of horizontal pixelline

The result is illustrated in Fig. 2.3. Here, a horizontal pixel line is extracted to detect the vertical edge of the material. In the corresponding graph, the y-axis represents the greyscale intensity, or Luma value, of the pixels along the x-axis. The transition from the material to the roller is clearly distinguishable, around the 300-pixel mark on the x-axis. This form of edge detection relies on the observation that edges typically occur at the boundaries between regions exhibiting significant intensity variations in Luma images.

To facilitate edge position detection, the derivative of the greyscale values is computed. Figure 2.4 displays the gradient for each pixel. The most prominent peaks indicate the locations of edges. Depending on the Luma intensity on either side of a boundary, the resulting derivative may assume either a positive or negative value. To support the identification of edge locations, it is common practice to consider the absolute value of the derivative. This approach simplifies peak detection by removing directional bias.

2.3.2 Increasing precision using subpixel edge detection

As previously mentioned, the efficacy of the edge detection algorithm in determining edge positions depends on its ability to accurately measure the size of individual pixels. These pixels serve as the fundamental unit of measurement in this method. However, the precision of this algorithm is inherently constrained by the dimensions of these pixels. The measurement precision can be enhanced by increasing the number of pixels representing the boundary. One way to achieve this is by positioning the camera closer to the measured edge. This approach, however, reduces the field of view (FOV). Al-

ternatively, increasing the resolution improves precision, but it also imposes a higher computational load. This is a limiting factor given the restricted processing power of the RPi version 4B.

To improve the accuracy of the captured and processed data without requiring hardware modifications, the Frequency Domain Zero Padding (FDZP) technique can be applied [21]. This method enhances the precision of the extracted data by increasing the resolution in the frequency domain. It allows the system to measure edge positions more precisely.

This edge detection method begins similarly to the basic approach described earlier. In this example, it achieves an initial edge detection precision of approximately 400 micrometres. Based on this detection, a second, more precise edge detection is performed using FDZP to further refine the edge position.

A Region of Interest (ROI) is defined symmetrically around the previously detected edge position. This section is then transformed into the frequency domain using a Fast Fourier Transform (FFT) [22]. Since the input data consists of discrete pixel values, the frequency representation is also discrete and inherently limited. This constraint arises directly from the Nyquist–Shannon sampling theorem. The theorem states that a discrete-time signal can represent only frequencies up to half the sampling rate, known as the Nyquist frequency [23].

Given that the input data is real-valued, the FFT output exhibits Hermitian symmetry [24], meaning:

- The first half of the FFT output represents positive frequencies.
- The second half is simply the complex conjugate of the first half, representing negative frequencies.

Since the second half does not contain additional information but is merely a mirrored version of the first half, it can be disregarded in further processing. However, a potential issue known as the Gibbs phenomenon [25] can introduce oscillations near the edges of the transformed data. To minimise these errors, trimming these edge values may be necessary. By consistently selecting a symmetrical ROI around the previously detected edge, these oscillations can be minimised.

Because the frequency representation is discrete, the spectrum has a finite resolution. This resolution can be enhanced through zero padding in the frequency domain, which increases the number of frequency bins without altering the original signal. This can be

seen as artificially increasing the sample frequency in the original signal. Although zero-padding does not introduce new information, it enables a perfect interpolation by refining the frequency spectrum. When the inverse FFT (IFFT) is applied to the zero-padded data, the result is a higher-resolution time-domain signal with improved precision.

Essentially, this technique subdivides pixels into smaller subpixels, enhancing edge detection precision. The ratio of subpixels to pixels is known here as the Factor 2.1. Use cases of this technique are presented in later chapters.

$$\text{Factor} = \frac{\text{\# subpixels}}{\text{\# pixels}} \tag{2.1}$$

However, despite its effectiveness, this approach has certain limitations. A key drawback is that the method is primarily suited for one-dimensional data. When extended to two-dimensional data such as images, the computational demand increases significantly due to the curse of dimensionality. This leads to an exponential growth in processing complexity [26].

Adhering to the Nyquist–Shannon sampling theorem is essential, particularly when analysing dynamic systems such as a material translating across a roller relative to a fixed camera. The theorem states that accurate signal reconstruction requires a sampling rate at least twice the highest frequency component present in the signal. Failing to satisfy this condition leads to aliasing, whereby high-frequency components are misrepresented as lower-frequency artefacts. This misrepresentation distorts the reconstructed signal. Ensuring compliance with the Nyquist criterion is therefore critical for obtaining precise and reliable measurements.

2.4 Width measurement process flow overview

The program is structured into two main phases: Preprocessing and Processing. During the Preprocessing phase, the program determines all preliminary values and constants required for width measurement. Since this phase is not time-sensitive, it allows for high-precision calculations that require more processing power. Once Preprocessing is complete, the Processing phase begins. This phase performs a continuous cycle of edge detections from both cameras to determine the material's width. In contrast to the non-time-sensitive nature of Preprocessing, Processing is highly time-sensitive. This sensitivity arises from the need to minimise cycle time, which is essential

for achieving a high number of measurements per unit time. Therefore, it is important to strike a balance between speed and precision.

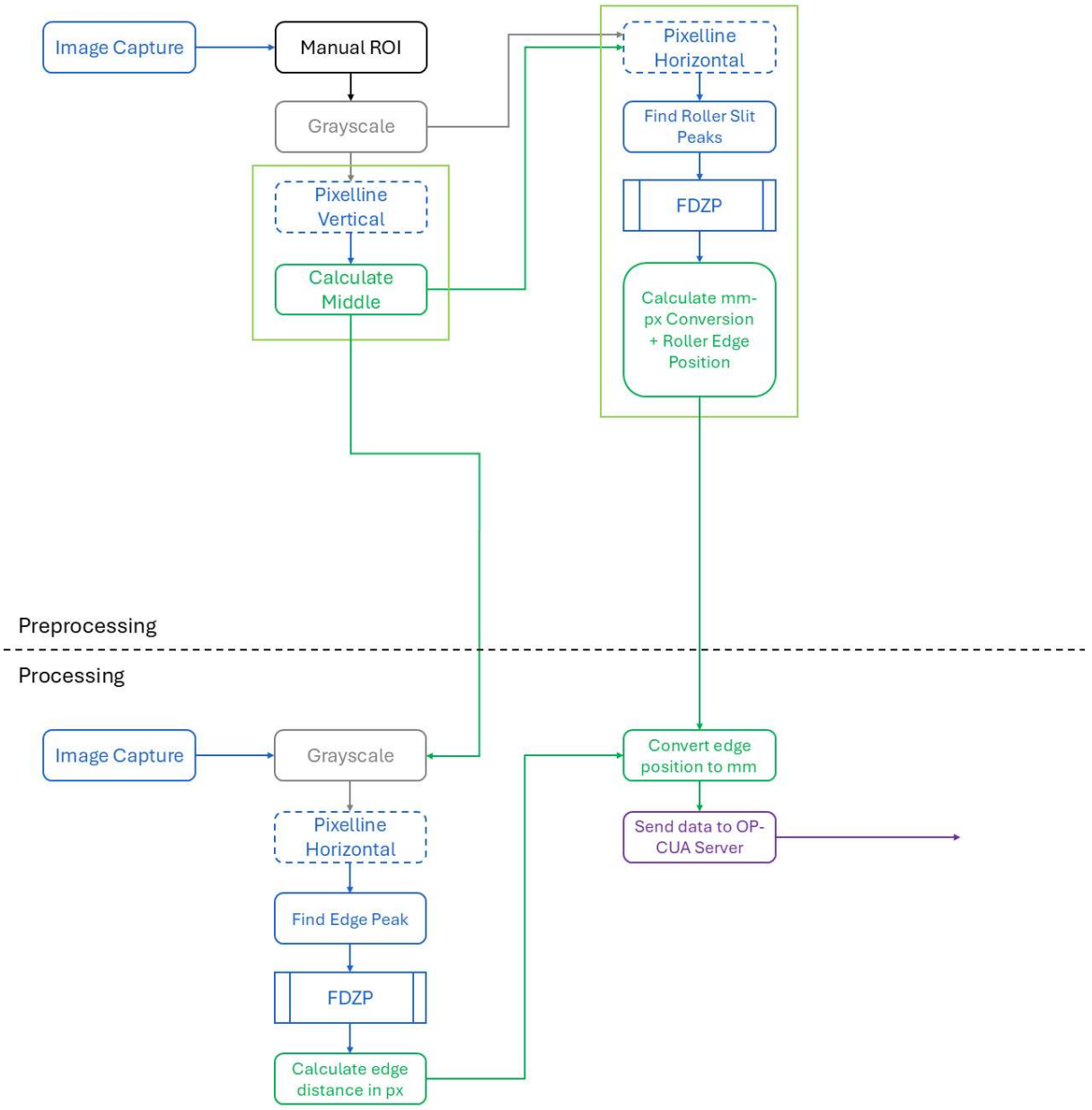

Figure 2.5: Process Flowchart

2.5 Preprocessing and image calibration

The entirety of the preprocessing is executed within a reduced, manually calibrated ROI of the complete image, as illustrated in Fig. 2.6.

Figure 2.6: Image of the preprocessing ROI of the roller

2.5.1 Choosing an effective resolution

When selecting an appropriate image resolution, it is important to balance image detail with processing time. Higher resolutions generally offer improved image quality but also demand more computational resources, which may reduce measurement efficiency. Edge detection under hardware constraints requires maintaining a relatively low cycle time to ensure a continuous data stream. To meet this requirement, systems typically operate at an intermediate resolution such as 640×480. This resolution provides a reasonable trade-off between image quality and processing speed. During the preprocessing stage, where accuracy takes precedence over computational speed, systems can employ a significantly higher resolution such as 4000×3008. This resolution also represents the maximum supported by the RPi High Quality lens, corresponding to 12 megapixels.

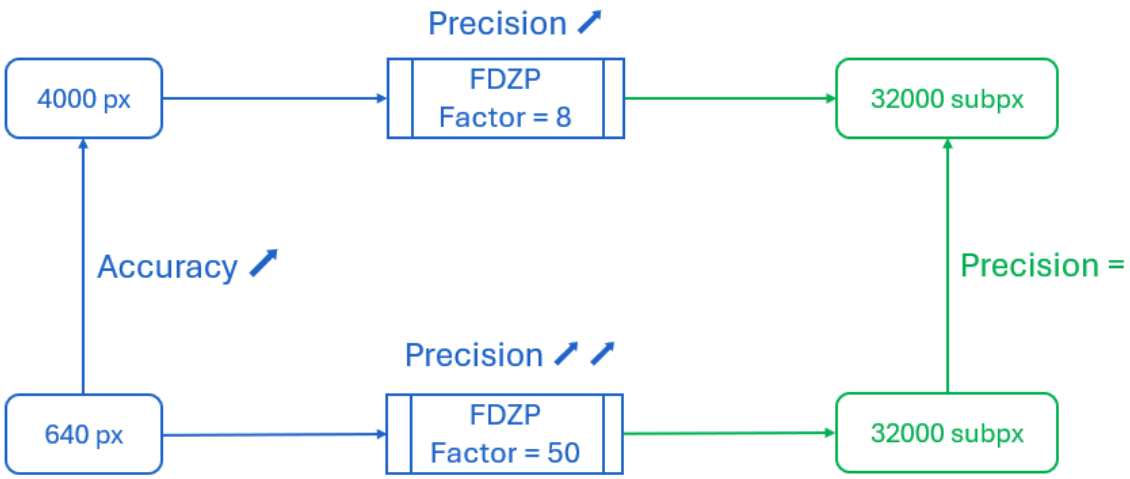

Figure 2.7: Resolution Influence on Accuracy and Precision

As illustrated in Fig. 2.7, increasing the resolution enhances measurement accuracy. Applying FDZP at higher resolution increases the pixel count within the ROI, which in turn reduces the Factor required to achieve the desired subpixel precision. Since the subpixel precision target remains constant, the higher pixel density reduces the need for an equally large scaling factor, as shown in 2.1. While the increase in precision, as indicated by the factor, is lower, the resulting subpixel precision remains unchanged. Although the precision stays the same, the accuracy improves.

2.5.2 Locating the roller middle

The initial step in the preprocessing stage involves determining the line along which edge detection will be performed. In this specific case, the optimal measurement location is at the middle of the roller. This position lies closest to the camera, which simplifies manual focus adjustments. Furthermore, measuring at this point ensures that edge detection remains constant and unaffected by variations in background colours. This contrasts with measurements taken at the roller's edge, which are susceptible to such variations.

To determine the middle of the roller in the image, a basic edge measurement is performed. This is achieved by analysing a vertical pixel line across the roller, as illustrated by the line in Fig. 2.8. From this line, a gradient is computed, as shown in Fig.

2.9.

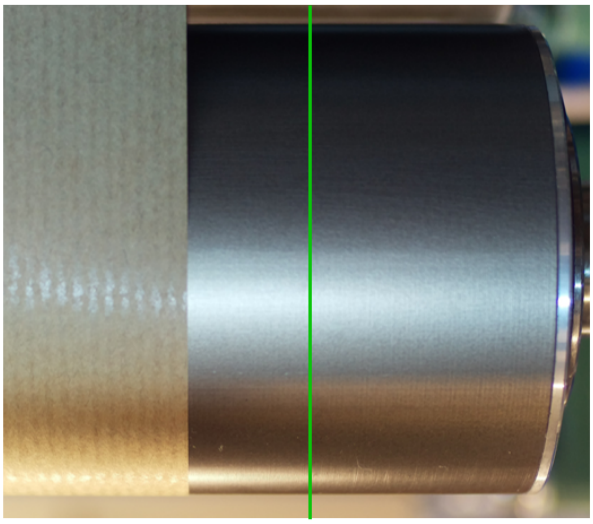

Figure 2.8: Vertical pixel line on the WPM roller

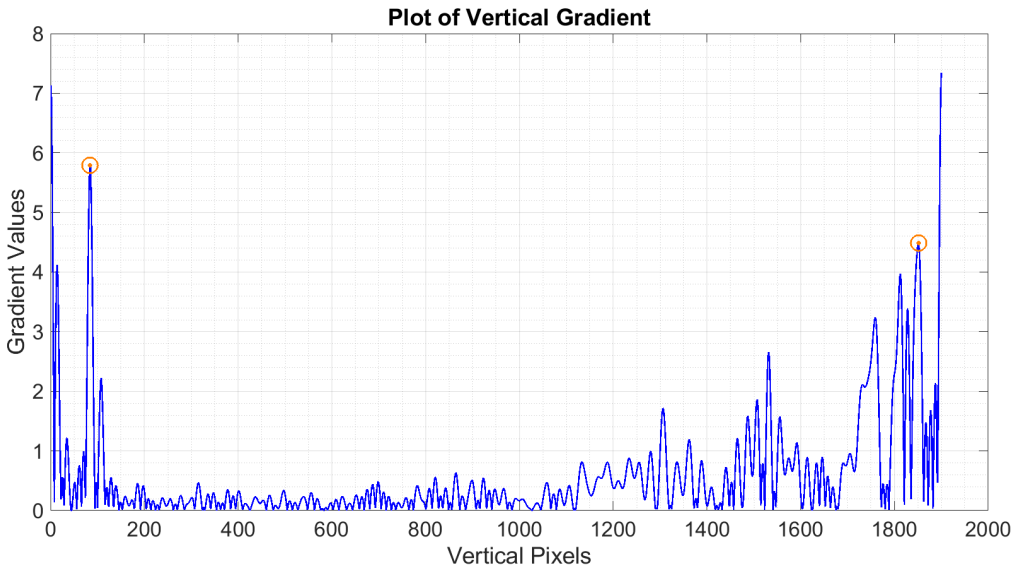

Figure 2.9: Vertical Gradient of vertical pixelline

The two edges of the roller can be identified based on the gradient. The positions of the detected edges are expressed in pixels. By summing these values and dividing by two, the middle of the roller can be determined. However, this value is relative to the image within the defined ROI. In this case, the middle is calculated based on the manually selected ROI. To determine the middle relative to the entire image, the system adds the pixel position of the upper edge of the ROI to the computed value.

It is important to acknowledge that this pixel position is derived from the high-resolution image used during the preprocessing stage. However, during the processing stage, the program employs a different, lower resolution. Consequently, this value must be converted to match the lower resolution, as pixel sizes differ between resolutions.

As previously mentioned, the middle of the roller is determined using a basic edge detection method, without applying FDZP. FDZP is used when measuring edges with a precision that exceeds the size of individual pixels. Since the middle of the roller serves as the reference for drawing the pixel line used in edge detection, any floating-point precision values are always rounded to the nearest integer.

2.5.3 Pixel to mm conversion ratio

Vision-based methods conventionally express distance measurements in pixels. However, these pixel values lack an inherent real-world reference. Consequently, differences across cameras generally prevent direct comparison without external calibration to ensure identical distances from the measurement plane. To enable meaningful interpretation, a conversion from pixels to a real-world unit is required. In this case, the chosen unit is millimetres.

Conducting a pixel-to-millimetre conversion requires measuring a known length in pixels using the camera. For this particular problem, it is possible to utilise multiple dimensions of the roller.

Conversion utilising the roller width

In the previous step during preprocessing, the system measured the top and bottom of the horizontal roller to find the middle, which it would subsequently use during the following phase of processing. These two points correspond to the width of the roller, which is a known measurement in millimetres. This approach is, however, flawed. As shown in Fig. 2.10

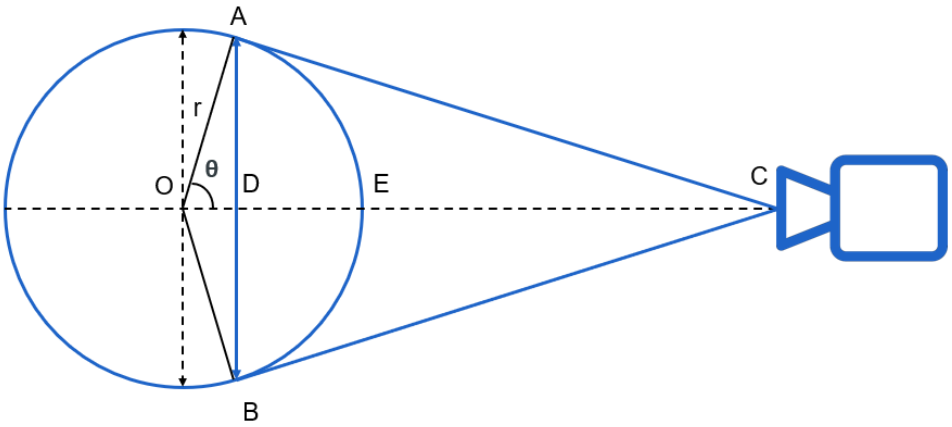

Figure 2.10: Roller width camera perspective showing potential tangential distortion

When observed by the camera, round objects such as the roller appear to have distorted dimensions due to the tangential camera view. Therefore, assuming the roller's observed width equals its diameter would be incorrect, as was previously calculated and documented [13].

$$\widehat{ODA} = \tan\theta = \frac{AD}{OD} = \frac{AD}{r\cos\theta} \Rightarrow AD = r\sin\theta \tag{2.2}$$

$$\widehat{OAC} = \cos\theta = \frac{r}{OC} \Rightarrow r = OC\cos\theta \tag{2.3}$$

Implementing 2.2 into 2.3:

$$AD = OC\cos\theta\sin\theta = \frac{OC}{2}\sin 2\theta \Rightarrow \theta = \frac{1}{2}\arcsin\frac{2AD}{OC} \tag{2.4}$$

$$r = OC\cos\left(\frac{\pi}{2} - \frac{1}{2}\arcsin\frac{2AD}{OC}\right) \tag{2.5}$$

With an expression for θ now established, substituting 2.4 into 2.3 yields a formula for the radius r as a function of the measured width of the object AB, where AB is defined as $2AD$. In 2.5, an argument correction is introduced by subtracting θ from $\frac{\pi}{2}$. The length OC represents the distance from the camera's focal point to the middle of the roller.

While this approach appears pragmatic in theory, the primary challenge lies in determining the value of OC. Since the objective of this edge detection method is to achieve a precision of ideally 50 micrometers or less, the pixel-to-millimetre conversion must be of comparable precision. Manually measuring the distance from the roller's edge to the camera is impractical. It is also unlikely to provide the required level of measurement accuracy and reliability. Finding a different strategy for this pixel-to-mm-conversion is a requirement for creating the stretch sensor.

Conversion utilising the roller slit

To mitigate the previously mentioned challenges, the stretch sensor utilises an alternative roller dimension to obtain a pixel-to-millimetre conversion: the roller slit. This feature consists of a 2-millimetre-wide chamfered edge located at the end of the roller, as illustrated in Fig. 2.11

Figure 2.11: Image of the right end roller slit

The primary benefit of measuring the roller slit is that it can be evaluated along the same horizontal axis as the edge detection. This axis corresponds to the middle line of the roller. This approach eliminates conversion errors arising from the tangential positioning of the camera relative to the roller, as previously discussed in the preceding

subchapter 2.5.3. The end of the roller functions as a fixed reference point during processing, against which the material position will be measured.

An additional benefit of using the slit measurement is the well-defined profile of the roller slit itself. As it is an integral part of the roller, it remains unaffected by background changes, with only variations in lighting conditions needing to be accounted for. Furthermore, the roller slit exhibits a well-defined and easily interpretable gradient, as illustrated in Fig. 2.13.

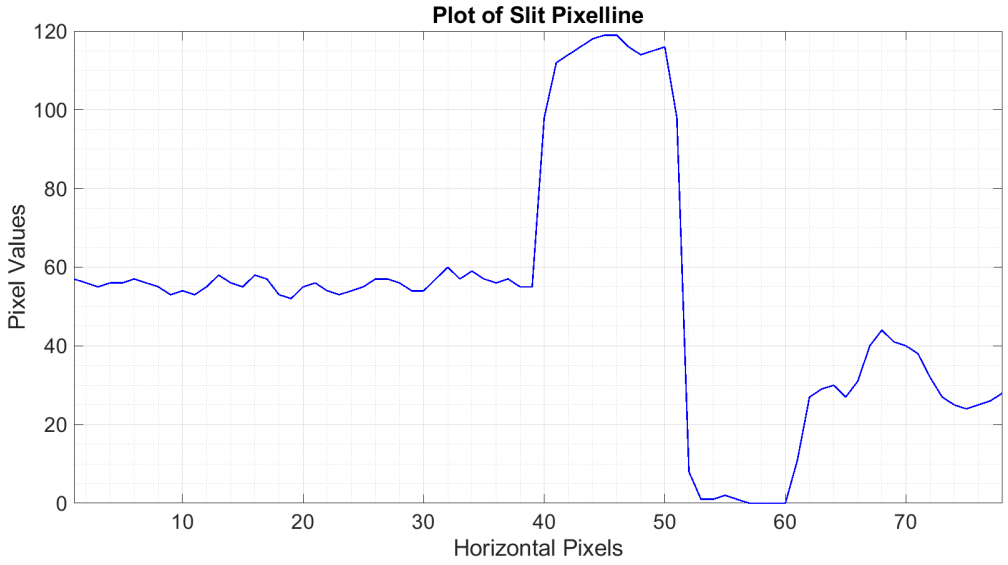

Figure 2.12: Horizontal pixelline of roller slit

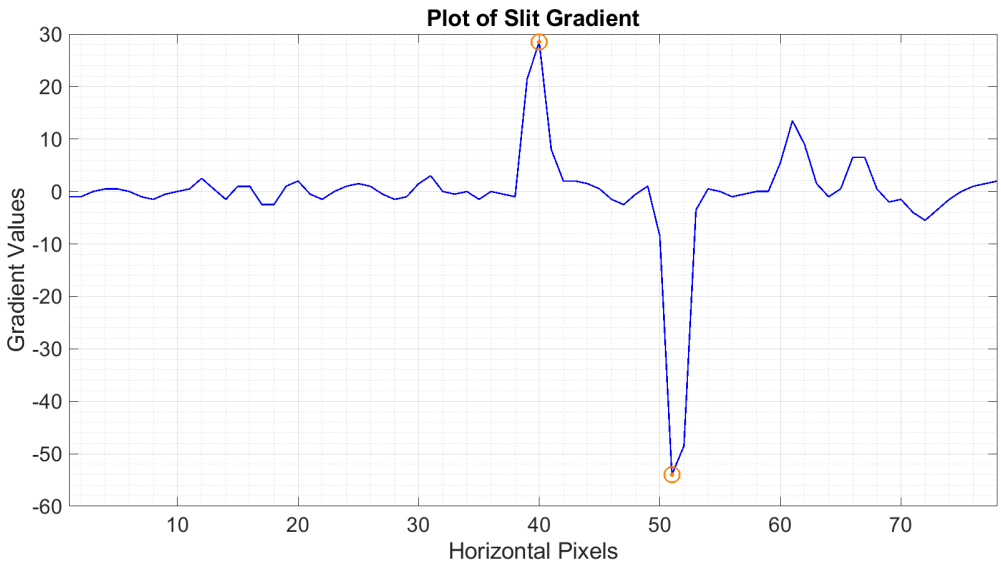

Figure 2.13: Pixelline gradient of roller slit

To ensure that the correct peaks are consistently identified by the program, the physical characteristics of the roller are first considered. The roller slit has a lighter material finish compared to the roller itself, while the black rubber end cap introduces a strong colour change. As a result, transitions at these locations are consistently captured in the Luma image and consequently appear in the gradient profile as well, largely independent of lighting conditions.

Upward ramps in the pixel line correspond to positive peaks in the gradient. These indicate a transition from a darker to a lighter Luma value, in this case, the roller slit. Conversely, the black rubber end cap causes a negative gradient, as it represents a transition to a darker region.

Based on this understanding, the following logic is applied. First, the program extracts all positive and negative peaks from the gradient. The tallest positive peak is assumed to correspond to the roller slit. Subsequently, the tallest negative peak located to the left of this positive peak is selected, corresponding to the end cap.

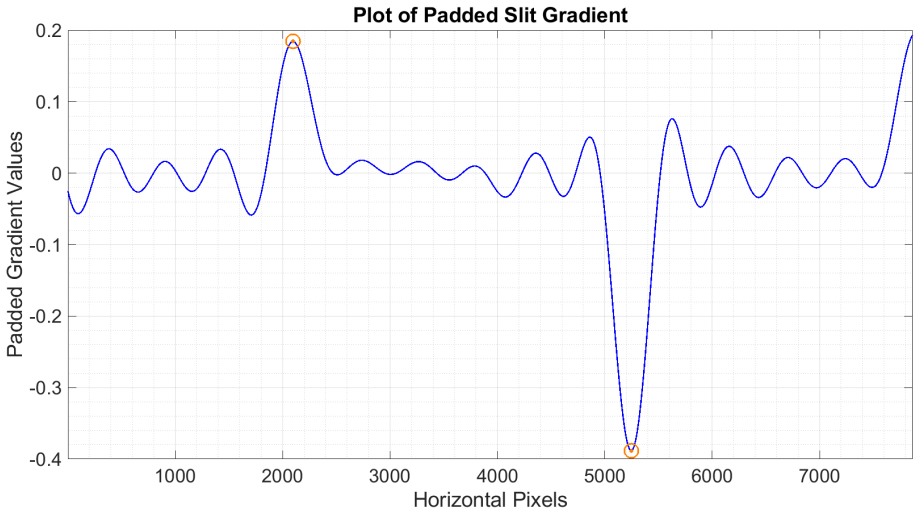

Figure 2.14: FDZ-padded pixelline gradient

With the slit peaks identified, FDZP can be applied to an ROI surrounding these peaks to determine the precise distance between the points in subpixels. This distance is then divided by the FDZP factor and added to the previously detected peaks, adjusted by subtracting the size of the ROI. The details of this conversion will be further elaborated in the processing chapter.

The distance of the roller slit in pixels can then be used to determine the pixel-to-millimetre conversion. The slit measures 2 millimetres. By accounting for the necessary conversion from high-resolution to low-resolution pixel measurements, one can establish the conversion factor. Since runtime constraints are less critical during preprocessing, this measurement can be repeated multiple times to compute a statistical average. This approach helps mitigate the impact of outliers and improves the accuracy of the conversion.

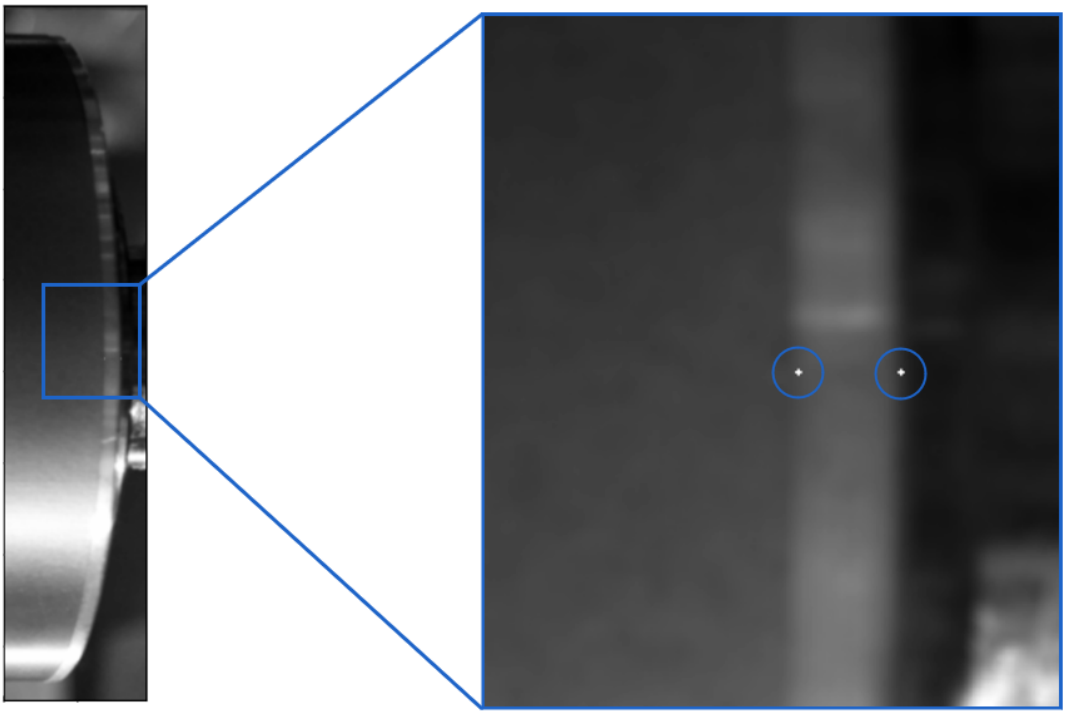

Figure 2.15: Roller slit detection on high resolution image with dots marking the measurement points

Figure 2.15 presents a zoomed-in view of the ROI, with white dots indicating the detected edges of the roller slit. These dots are one pixel wide, illustrating the level of precision achievable with the high-resolution image. While the actual edge detection process using FDZP provides even greater subpixel precision, this cannot be visually represented in the image.

Measurement imperfections

Although the roller slit for conversion serves to eradicate errors that the camera's tangential viewing angle incites, it is important to acknowledge that alternative sources of error also exist. When measuring distances along straight lines near the periphery of a camera's FOV, inaccuracies arise due to the inherent limitations of the pinhole camera model [27]. This model assumes that light rays pass through a single point before being projected onto a flat image sensor. However, this simplification introduces perspective distortion and radial distortion, particularly towards the edges of the image.

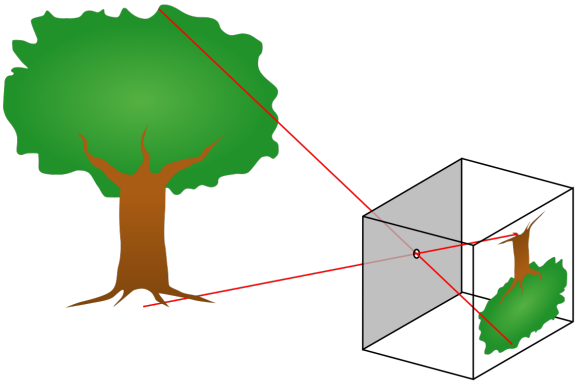

Figure 2.16: Example of pinhole camera model projection

Perspective distortion

In the pinhole model, objects positioned farther from the optical axis appear smaller due to the convergence of light rays at the focal point. This effect is more pronounced at the edges, where the projection angle is steeper, leading to stretching or compression in distance measurements [28].

Radial distortion

Certain lenses introduce barrel distortion, while others produce pincushion distortion, both of which cause straight lines to appear curved as they extend outward from the image centre, as illustrated in Fig. 2.17 [29]. As a result, equal distances along a real-world straight line are not uniformly projected onto the image plane, distorting measurements at the periphery. [30]

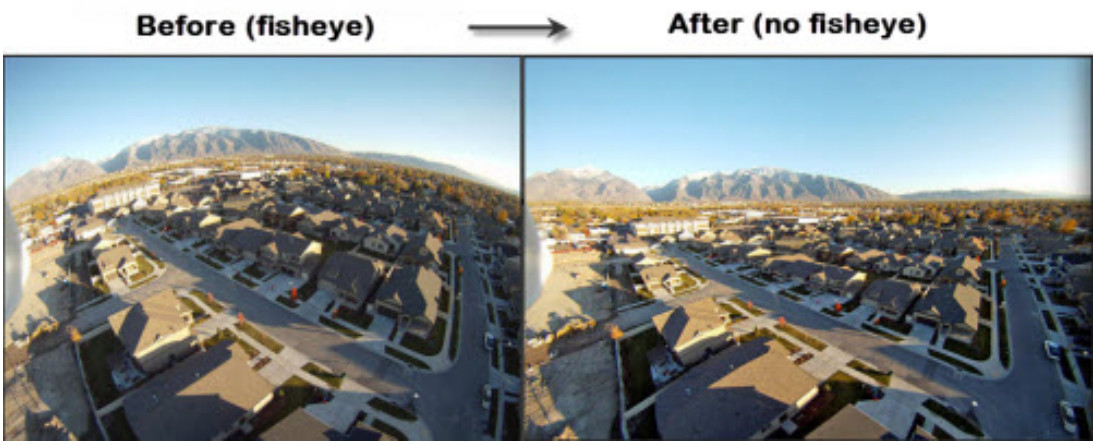

Figure 2.17: Radial distortion correction: Gopro fisheye

Intrinsic camera calibration

The roller slit is positioned at the end of the roller. Because the sensor must capture as much of the roller as possible within its FOV, the slit measurement will invariably occur at the edge of the camera's field of vision. Relocating the camera after preprocessing is not advisable, as it will inevitably affect the accuracy of the pixel-to-millimetre conversion.

These errors, however, can be mitigated through intrinsic camera calibration. This process estimates the internal parameters of a camera, including its focal length, principal point, and lens distortion coefficients. By performing this calibration, radial distortion and perspective errors can be corrected. This leads to more accurate image-based measurements [31].

Intrinsic calibration determines the camera matrix, which contains the focal lengths and the optical centre. This enables the rectification of perspective distortion, ensuring that measurements remain consistent regardless of the object's position within the FOV. Additionally, intrinsic calibration identifies radial and tangential distortion coefficients, which quantify how straight lines may appear curved or skewed, particularly near the edges of the image. Both matrices are written down in Appendix 4.9. By applying these coefficients, Python libraries such as OpenCV [32] can perform an undistortion transformation, straightening lines and correcting the stretching effect observed at the periphery.

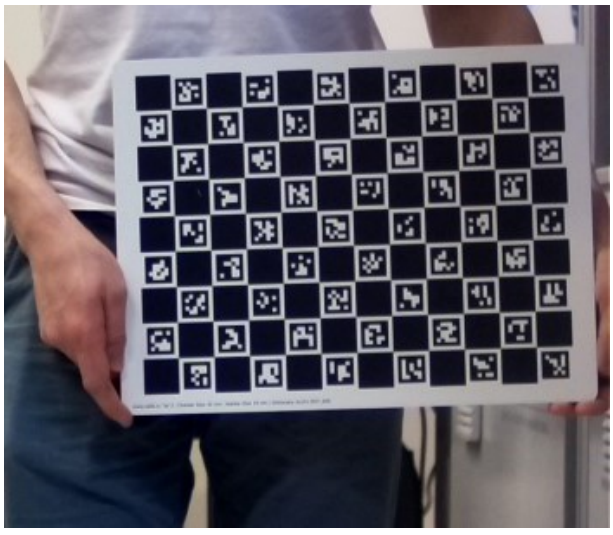

Figure 2.18: ChArUco board

In practice, intrinsic camera calibration can be performed using a chessboard, a checkerboard, or a ChArUco board, as shown in Fig. 2.18. A ChArUco board is a hybrid of a checkerboard and an ArUco marker pattern, similar to AprilTags. It combines the benefits of sub-pixel corner detection of the chessboard with robust marker identification, enhancing both accuracy and reliability in the calibration process.

Conversion utilising a roller marker

The initial development of the roller slit measurement was undertaken early in the design of this stretch sensor. However, it was later discovered that the FOV of the roller was too narrow to effectively track the material over a sufficiently long section of the roller. The solution to this issue was to reposition the cameras further away from the roller, thereby increasing the field of view. The FOV increased from 10 cm initially to 22 cm and 18 cm for the right and left cameras respectively. The difference between the two cameras arises from the fact that the material used during testing was not centred on the roller. As a result, the right camera required a wider view of the roller to accurately measure the distance to the material's edge.

While this adjustment resolved the visibility constraint, it introduced a new challenge. Although measuring the roller slit had previously been an effective approach, the increased distance rendered it difficult to measure reliably. This resulted in frequent outliers and reduced robustness under varying lighting conditions. To address this, a printed paper marker was applied to the roller. This marker, with a known width of 10 millimetres, five times the width of the roller slit, offers significantly improved contrast against the roller, enhancing measurement reliability.

Despite this change, the end of the roller will still serve as the fixed reference point during processing.

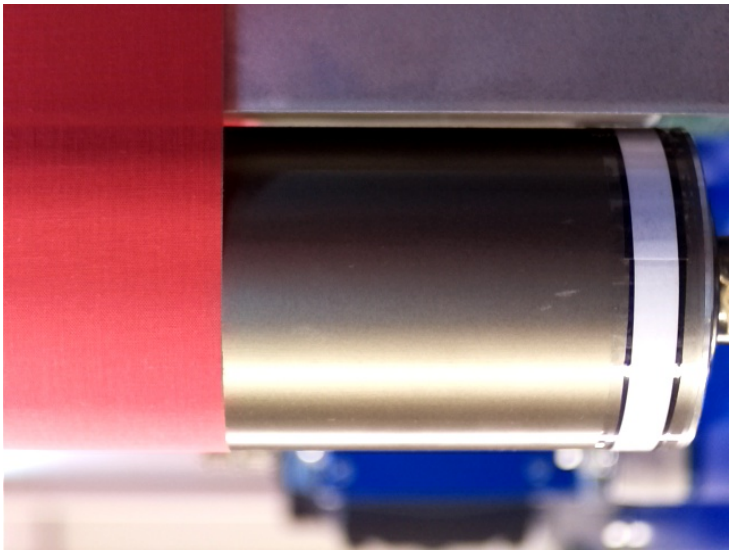

Figure 2.19: Roller with white preprocessing marker

Although the use of a conversion marker may somewhat deviate from the original vision of developing a machine-independent stretch sensor, it remains a simple and minimally invasive modification to the roller. This addition significantly enhances the reliability and accuracy of the pixel-to-millimetre conversion, thereby improving the overall quality of the measurements obtained with this sensor.

2.6 Real-time processing

Once preprocessing has been completed and the resulting variables of middle, conversion, and roller end have been found, the actual edge detection process can commence. The processing phase encompasses the entire workflow, including image capture, edge measurement, pixel-to-millimetre conversion, and the storage or transmission of data to a file or server. This process operates in a continuous loop until the program is manually stopped or the RPi is powered down.

2.6.1 Edge detection workflow

Edge detection is the primary algorithm executed during processing. The process begins by capturing an image and converting it to greyscale. Using the previously determined middle from the preprocessing stage, a horizontal pixel line is extracted, from which the gradient is computed.

The workflow, shown in Fig. 2.5, used to detect the edge follows the same approach as previously described in Section 2.3.2. This method, initially utilised during preprocessing to measure the roller slit for pixel-to-millimetre conversion, is now applied during processing to ensure precision in edge detection.

First, the middle line is taken from the low-resolution image. This pixel line is shown in Fig. 2.20. From this pixel line, the gradient is shown in Fig. 2.22.

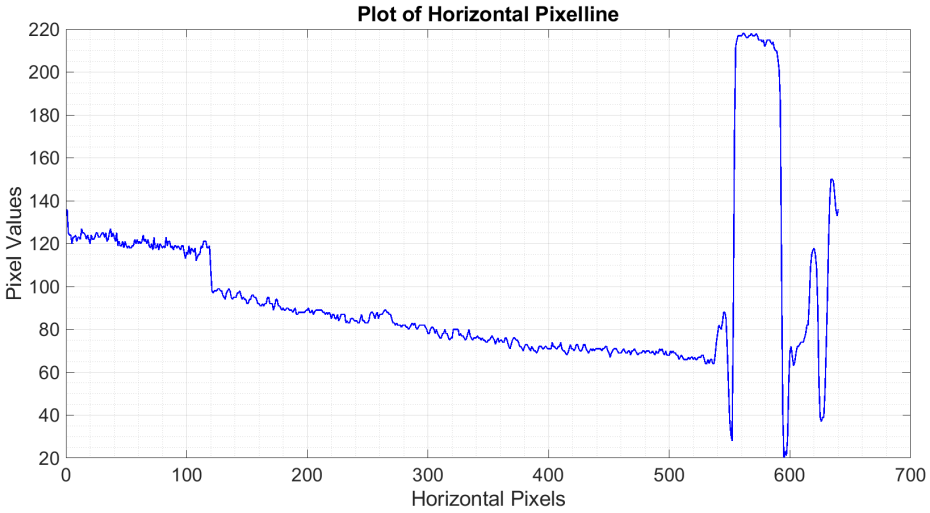

Figure 2.20: Horizontal pixelline for material edge detection

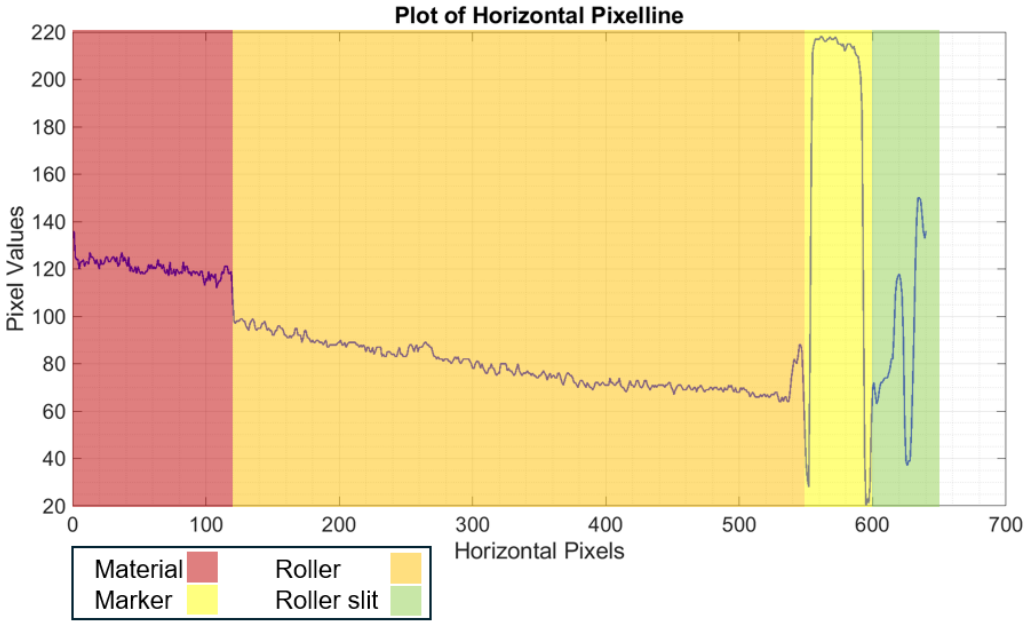

Figure 2.21: Horizontal pixelline (Highlighted sections)

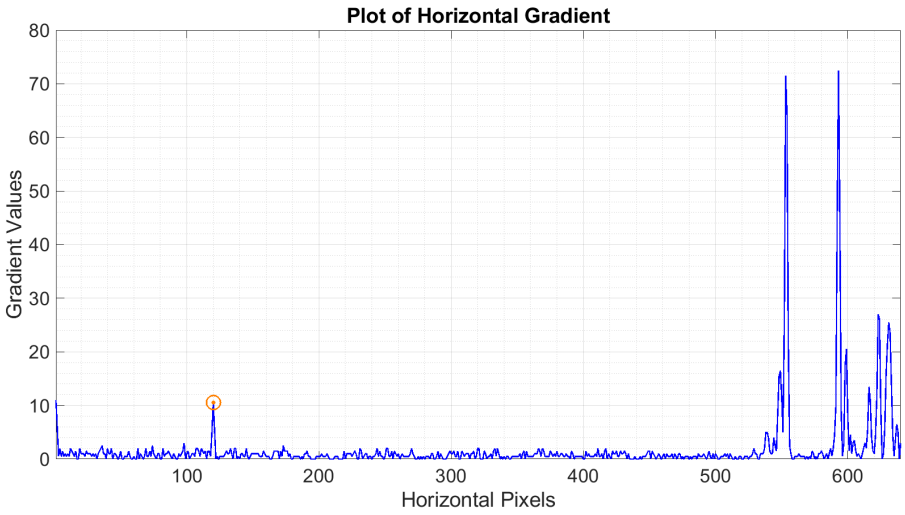

Figure 2.22: Horizontal pixelline absolute gradient

As illustrated in this image, the peak at approximately 120 pixels signifies the required edge detection boundary. However, on the right, two additional larger peaks are present, which result from the white marker on the roller. One of the most critical aspects of developing a reliable edge detection method is ensuring that the program consistently detects the boundary between the material and the roller, rather than mistakenly identifying the marker or the roller slit.

In this case, since the position of the marker relative to the camera is known and remains constant while the program is running, it can be excluded from the search area using a fixed parameter. By applying an offset of 120 pixels from the right side of the line, the marker is effectively ignored. However, this simple solution also introduces a limitation: if the material moves inside the excluded area, the program will no longer be able to track the material edge.

These limitations can be addressed by introducing additional rules to the peak selection logic. Rather than simply identifying the highest peak within the defined search area, all peaks exceeding a certain threshold can be selected. When applied to the gradient, this selection process is performed over a region starting at 0 pixels and extending to the edge of the roller. This is done specifically up to the previously identified roller slit from preprocessing. This results in three possible scenarios:

- Three peaks are detected: Two correspond to the marker, meaning the material is not overlapping the marker. Edge detection should proceed using the third peak from the right.

- Two peaks are detected: The material is partially overlapping the marker. Edge detection should proceed using the second peak from the right.
- Only one peak is detected: The material is completely obscuring the marker, and edge detection can proceed using the detected peak.

While this method eliminates the need to manually set a parameter to exclude the marker, it introduces the requirement to define a threshold above which peaks are detected. Without an appropriate threshold, peaks caused by noise or light reflections could be mistaken for the material edge. To prevent this, a sufficiently high threshold must be selected.

Different types and colours of material will result in varying peak heights on the gradient line because of their different Luma values. This means the threshold must be somewhat material-specific. Additionally, this method has the drawback of making it difficult to detect when the material is no longer within the camera's FOV. In such cases, two peaks may still be detected, one from the roller edge and one from the marker. This may lead the system to incorrectly infer that the material edge has suddenly shifted close to the roller edge, effectively over the marker. For these reasons, and given that in this specific case of the WPM the material generally does not approach the marker, the original method using an exclusion parameter is retained.

Now that the material peak has been selected, a symmetrical ROI can be taken around it, as shown in Fig. 2.23. FDZP can then be applied to determine the precise edge of the material, as seen in Fig. 2.24. To mitigate the Gibbs phenomenon, 2% of either side of the ROI is removed after FDZP has been applied.

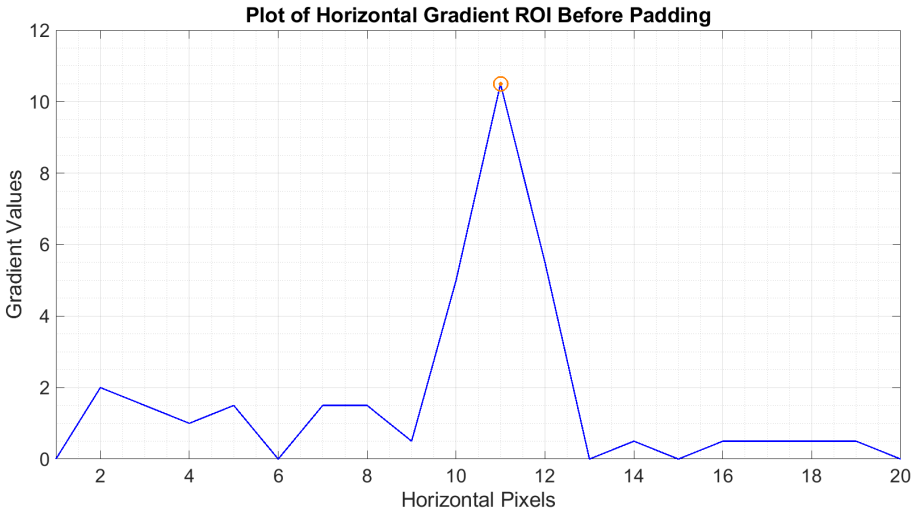

Figure 2.23: Horizontal absolute gradient ROI before padding

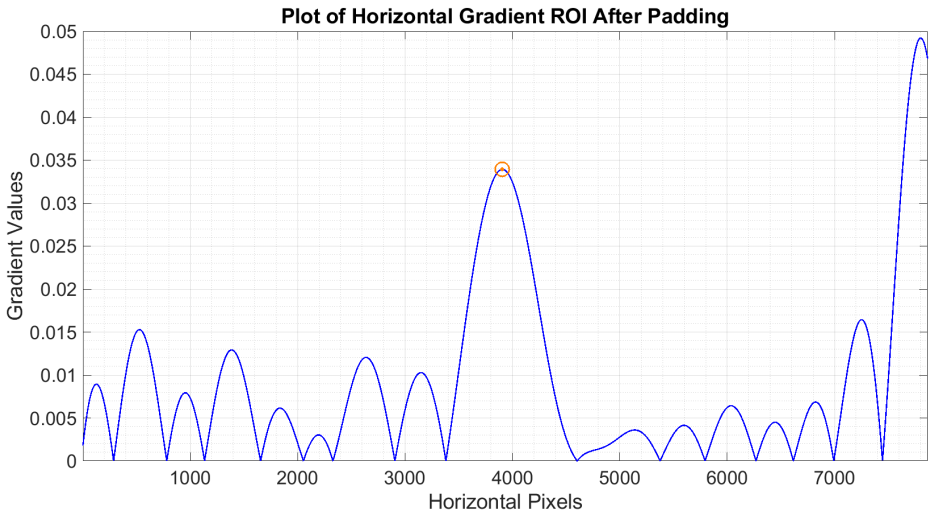

Figure 2.24: Horizontal absolute gradient ROI after padding

Now that the values of the basic and FDZP-enhanced edges have been determined, the distance in pixels can be calculated. It is important to note that these values cannot simply be added together, as both measurements are obtained in different reference frames and use different units. Figure 2.25 provides a schematic overview of the required conversion.

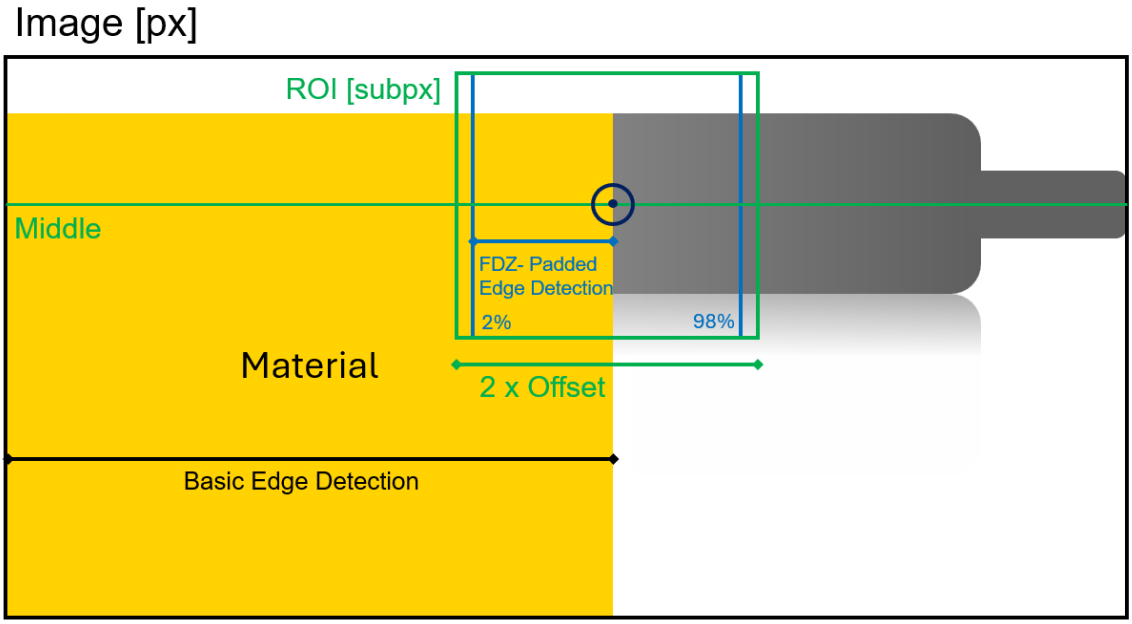

Figure 2.25: Edge detection conversion schematic overview

The basic edge detection takes the left edge of the image as its reference and is measured in pixels. The ROI is positioned symmetrically around this detected edge and has a width of twice the offset (a chosen parameter, which in this case is 10 pixels).

Selecting an appropriate offset size is important. If the ROI is too small, the detected edge will be poorly defined. Conversely, if the ROI is too large, it diminishes the effectiveness of FDZP, as it reduces the ratio of subpixels to pixels, referred to here as the Factor. Maintaining the same level of precision under these conditions would require an increase in the subpixel accuracy target, which in turn would lead to a higher computational load.

The reference for the FDZP-enhanced edge detection is the ROI, with an additional 2 percent removed on either side to mitigate the Gibbs phenomenon. This measurement is expressed in subpixels. To convert it back to pixels, the subpixel value must be divided by the Factor.

Once both measurements have been adjusted to the same reference frame, they can be combined. However, since the offset has been accounted for twice, it must be subtracted from the final result. Equation 2.6 presents the complete calculation used for this process.

$$\frac{\text{FDZ-Padded Edge Detection} + 0.02 \times \text{Offset}}{\text{Factor}} + \text{Basic Edge Detection} - \text{Offset} \quad (2.6)$$

This value is then multiplied by the pixel-to-millimetre conversion factor to achieve the final distance in millimetres. Figure 2.26 shows the result of this edge detection.

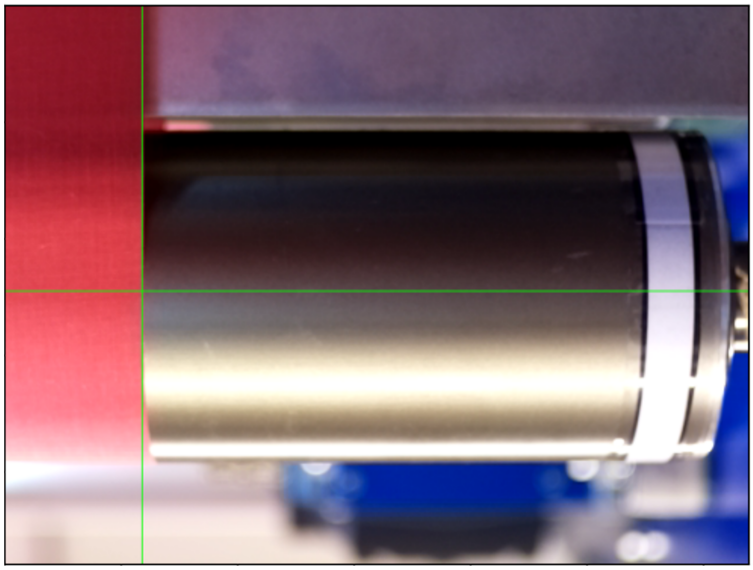

Figure 2.26: Material with Edge Detection highlighted by green pixelline

2.6.2 Cycle time enhancements

In addition to the precision of the stretch sensor, another crucial aspect is its cycle time. A higher cycle time results in fewer measurements per second. This leads to a lower data density and reduced responsiveness for systems relying on the generated stretch data.

By default, the prescribed edge detection method using FDZP is computationally intensive, often limiting the measurement frequency to approximately 10 Hz, with fluctuations ranging between 5 and 25 Hz. One potential way to mitigate this issue is by reducing the target subpixel amount and, consequently, the FDZP Factor. However, this comes at the cost of reduced precision. Therefore, an alternative approach is required to achieve a balance between precision and computational efficiency.

Pooling

The processing steps for edge measurement are structured within an infinite main loop, where each step is executed sequentially. However, certain tasks have the potential to be performed concurrently. To optimise execution, these tasks are dynamically distributed across the 4 CPU cores of the RPi. This enhances efficiency and reduces overall processing time. By leveraging parallel execution, this approach minimises idle time and ensures optimal management of computational resources.

In this context, pooling operates by establishing a worker pool, a collection of workers that handle tasks asynchronously as they arrive. This mechanism adheres to the producer-consumer model, where tasks are enqueued and retrieved by available workers for independent execution [33]. Python's multiprocessing module facilitates process-based parallelism. This allows it to execute CPU-bound tasks concurrently by distributing workloads across multiple processor cores.

By dynamically assigning tasks to workers instead of processing them sequentially, CPU utilisation is maximised, particularly when tasks arrive at varying rates [34]. This technique is especially beneficial for embedded systems such as the RPi, where real-time constraints necessitate optimised execution cycles [35]. Moreover, research in parallel computing has demonstrated that pooling can significantly reduce execution bottlenecks in systems with limited processing power by ensuring that it balances the workload.[36].

As a direct consequence of this optimisation, the program's cycle frequency increases to approximately 25 Hertz, ensuring that the sensor operates at the required frequency. By effectively distributing computational tasks and enabling asynchronous execution where possible, pooling enhances both system responsiveness and throughput.

Duplicates and slow pooling

During the implementation of pooling, the program identified an issue. Consecutive entries in the output file occasionally contained identical values, despite performing calculations with a precision of 18 decimal places. These duplicate values compromise data quality, which reduces the system's responsiveness to measurement variations. This can also potentially introducing inconsistencies during validation in a later section.

This anomaly is likely attributable to race conditions or data overwriting within a concurrent execution environment. Specifically, if multiple worker threads or processes share

a common variable without adequate isolation, one task may overwrite another's intermediate results. This can happen before they are committed to the output file or transmitted to the server. Additionally, if variable updates are not properly synchronised, multiple workers may attempt to modify the same variable simultaneously, leading to duplicated or erroneous values.

To mitigate these issues, it is essential to ensure that variables are updated in a controlled manner before being accessed in subsequent processing steps. This can be achieved by enforcing synchronisation mechanisms, such as ensuring that critical processes complete before data is transmitted to the server. By implementing these measures, variables remain consistent and correctly updated, preventing erroneous duplication in the output.

Although this correction effectively resolves the issue of duplicates encountered in the previous pooling implementation, it introduces a new bottleneck between the calculation and writing phases. Consequently, this revised approach will be referred to as "Slow Pooling", in contrast to the original "Fast Pooling" method. Slow Pooling results in a cycle time of approximately 18 to 22 Hertz. While this represents an improvement compared to the absence of pooling, it remains below the potential performance achievable with the RPi version 4B hardware.

Multithreading

To further enhance performance, the image acquisition and processing steps were decoupled and executed in separate threads using Python's multithreading module. Initially, the system operated in a strictly sequential manner, where an image was captured, processed, and transmitted to the server before the next iteration could commence. However, this approach introduced significant delays, as the camera remained idle during computation.

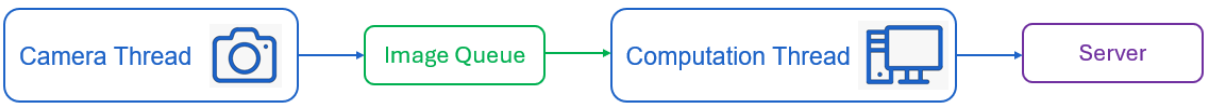

Figure 2.27: Schematic showing process flow with grouped multithreaded tasks

To overcome this limitation, a producer-consumer model was implemented, illustrated in Fig. 2.27. The image acquisition process was assigned to a dedicated thread that

continuously captures frames and stores them in a queue. Simultaneously, a second thread retrieves images from the queue, performs edge measurement calculations, and transmits the results to the server. This architecture ensures uninterrupted image capture while processing occurs concurrently, thereby minimising downtime and improving overall throughput. By enabling efficient task scheduling and resource utilisation, this approach prevents computational and I/O-bound operations from interfering with one another.

Additionally, in cases where the volume of data being transmitted to the server is substantial, a third thread can be introduced specifically for data transmission. However, for the stretch sensor, where only single floating-point values are sent, this was deemed unnecessary.

This implementation, when combined with the Slow Pooling approach, achieves a cycle time approaching 30 Hertz. This represents the hardware limit for the RPi version 4B and camera module setup, and finally realises the full potential of the available hardware.

However, careful handling is required when working with separate threads. It is entirely possible for one thread to operate significantly faster than another. If the camera thread cycles more rapidly than the computation thread, a backlog may develop in the image queue. As the queue fills, this backlog introduces latency between image capture and edge measurement. To mitigate this, the queue size is capped at five images, and the lifetime of each image is monitored. If an image exceeds a predefined age threshold, it is discarded to prevent outdated data from affecting system performance.

2.6.3 Server communication between PLC and RPi

Once the edge measurement is completed, the resulting value is stored as a floating-point number. It is then sent to a database before the next measurement cycle begins. This communication is facilitated via Open Platform Communications Unified Architecture (OPC UA), a machine-to-machine communication protocol widely used in industrial automation for secure and reliable data exchange [37]. By utilising a Python library, the RPi can establish an OPC UA client instance that connects to the specified server [38].

The communication between the RPi and the server is established using the Transmission Control Protocol / Internet Protocol (TCP/IP) [39]. TCP/IP functions as the transport mechanism that enables the RPi to interact with the server over the network via the OPC UA protocol. Within this framework, the Python library encapsulates OPC

UA messages into TCP packets, which are then transmitted through the broker network of the WPM to reach the server. TCP plays a crucial role in ensuring reliable data transfer by managing packet sequencing and initiating retransmissions in the event of errors.

This setup enables the RPi to read from or write data to a designated node on the server, ensuring seamless data exchange between the two devices. In this case, the server is hosted on a PLC inside the WPM [40]. The transmitted data can be accessed and utilised in subsequent practical applications.

Synchronisation error

As previously mentioned, the stretch sensor consists of two RPi version 4B units, each equipped with a camera module. These RPi devices execute identical code, with the sole exception that one of them mirrors the captured images. They operate entirely independently and transmit their data to the server as soon as it becomes available.

Upon receiving these values, the server subtracts them from 670 millimetres, corresponding to the roller's length. The server reads the data from the nodes at fixed intervals, which do not necessarily coincide with the exact moments the measurements are taken. Consequently, a synchronisation error may arise between the two values used to determine the material's width, as the server always processes the most recently received value. Figure 2.28 provides a schematic representation of how this error can be interpreted.

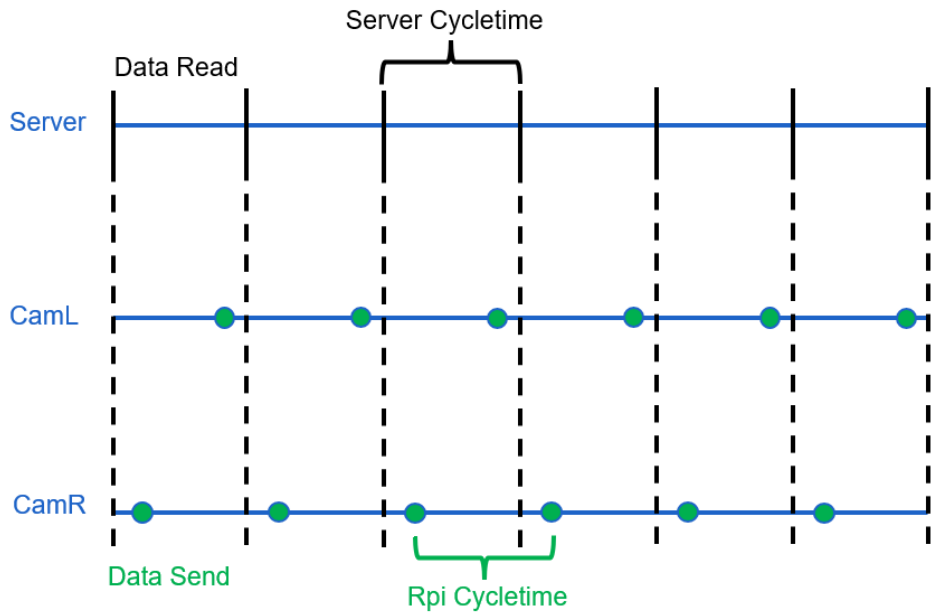

Figure 2.28: Synchronisation error schematic example

In this example, the server's cycle time is assumed to be identical to that of the RPi units. However, in practice, the server operates at a significantly higher cycle rate, resulting in multiple readings of the same data before a new measurement is available. Consequently, the effective cycle time of the stretch sensor should be defined by the cycle time of its limiting factor, which, in this case, is the RPi units.

The temporal offset between data transmissions from CamL and CamR can be considered random, as it is entirely dependent on the relative timing of when each program was initiated. However, in the worst-case scenario, the synchronisation error could reach the full duration of the cycle time of the edge measurement program.

The cycle time of the RPi fluctuates around 30 Hertz. At this frequency, for a typical material speed in the WPM, such as 30 metres per minute, the maximum possible material displacement between the two measurements from CamL and CamR is approximately 1.667 centimetres. This displacement represents the distance the material can travel between two consecutive measurements. In practice, however, this error is expected to be lower, as discussed previously. This synchronisation error can be further reduced by either decreasing the material speed or increasing the measurement frequency by reducing the cycle time of the program. However, achieving a lower cycle time would likely necessitate hardware upgrades.

An alternative approach to eliminating this synchronisation error entirely is to establish synchronisation between the RPi units. The most straightforward method would be to transmit data at a fixed, predetermined time. However, a fundamental limitation of this approach is that the RPi version 4B lacks an internal Real-Time Clock (RTC), which would otherwise facilitate precise timing.

Another potential solution is to implement a request–receive model, whereby the RPi units transmit data only when explicitly requested by the server. In this configuration, the RPi operates as a Modbus server. The PLC sends requests to the RPi at predefined intervals, which are then processed and responded to accordingly. Modbus TCP proves particularly advantageous in setups where the RPi handles custom logic or interfaces with external systems [41]. This model facilitates structured communication while maintaining compatibility with industrial protocols commonly used in automation environments. This method reduces the synchronisation error to the communication delay between the RPi and the server, which is typically in the microsecond range. However, this approach introduces additional communication overhead, which may, in turn, increase the cycle time of the edge measurement process, potentially affecting overall system performance.

Due to these disadvantages, no synchronisation correction has been implemented in the stretch sensor. However, testing at lower, more commonly used speeds in the WPM can further reduce this error. While the fundamental proof of concept remains unchanged, a more industry-ready version of the sensor may require slight hardware upgrades. For instance, the RPi version 5, with its cycle time of up to 60 Hertz, would already halve the maximum synchronisation error, making it a viable improvement for industrial applications.

2.7 Stretch sensor performance validation

Following the completion of the stretch sensor's software development, a series of experimental evaluations were undertaken to determine its industrially relevant characteristics. These assessments aimed to establish key performance metrics and to identify optimal operating points for various configurable parameters. This to ensuring the sensor's suitability for practical deployment in industrial environments.

The key characteristics that must be determined and benchmarked against existing sensor solutions are the cycle time and precision of the stretch sensor. The precision was evaluated in accordance with the ISO 5725-1:2023 standard, "Accuracy (trueness

and precision) of measurement methods and results" [42], thereby ensuring alignment with internationally accepted industrial practices. The validation process begins by assessing each camera individually in its role as an edge detection sensor. Following this, the system is evaluated in its entirety to verify its performance as a stretch sensor.

2.7.1 Selecting the optimal operating point

To determine the most favourable industrial characteristics of the stretch sensor, it is necessary to identify the optimal operating point. During testing with various scaling Factors for the FDZP technique, the tests observed that increasing the factor led to a corresponding growth in data size. This increase in data volume theoretically improves measurement precision; however, it also imposes a greater computational burden. As a result, higher FDZP factors were found to significantly extend the cycle time of the edge detection algorithm, while offering diminishing returns in terms of actual measurement precision.

These findings suggest that an optimal operating range exists for the FDZP technique, one that achieves a favourable balance between high theoretical precision and acceptable cycle time. This balance, in turn, maximises overall efficiency.

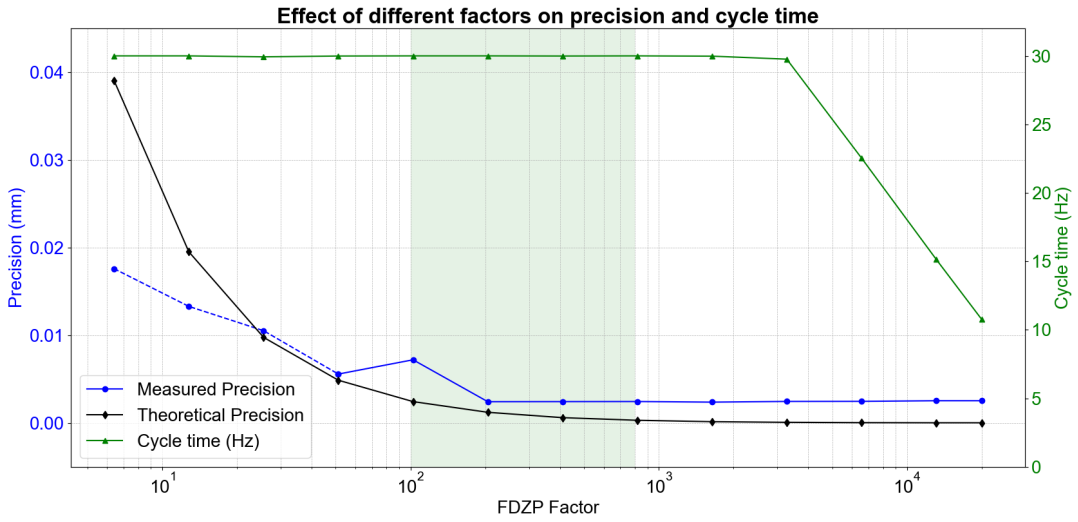

Figure 2.29: Optimal working region for edge detection at varying FDZP Factors (CamL)

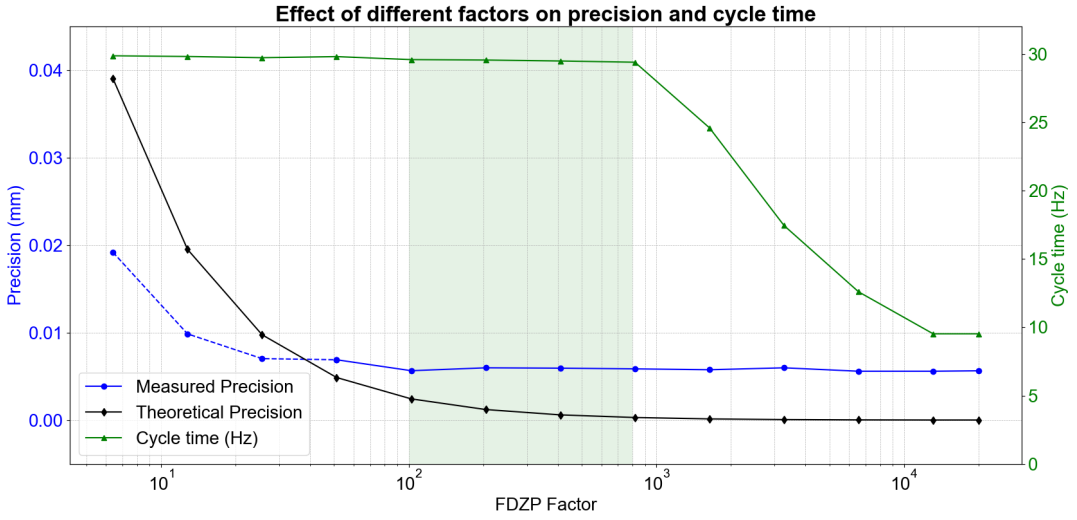

Figure 2.30: Optimal working region for edge detection at varying FDZP Factors (CamR)

Figures 2.29 and 2.30 present the results of a series of static tests conducted using different FDZP Factor values. In accordance with the ISO 5725-1:2023 standard, the precision of a sensor is assessed by repeatedly measuring the same value, or a steady-state condition of the sensor. Within the WPM environment, this was achieved by halting the machine while maintaining the material under tension to ensure a stable reference. The standard deviation (SD) of the resulting measurements reveals the precision of the sensor.

The x-axis displays a logarithmic scale for the FDZP factor. By multiplying the sub-pixel ratio with the pixel-to-millimetre conversion factor, the theoretical precision can be calculated. It is evident that an increase in the factor yields an improvement in theoretical precision (i.e., a lower numerical value). However, it can be observed that the measured precision does not attain the theoretical precision at medium to high Factor values. This discrepancy is primarily attributed to the inherent uncertainty present at the physical edges of the material. In practice, material edges are not cut with extreme precision, which introduces a degree of variability and thus slightly reduces measurement precision. Additionally, the lenses employed in this setup require manual adjustment and tuning of the focus. Unless this focus is set with absolute precision, a blurring effect may occur at the material boundary, further contributing to uncertainty in the edge detection process. This effect also largely accounts for the difference in precision observed between CamL and CamR. It is highly likely that minor differences in manual focus adjustment and slight variations in the distance between each camera and the

roller have contributed to this divergence.

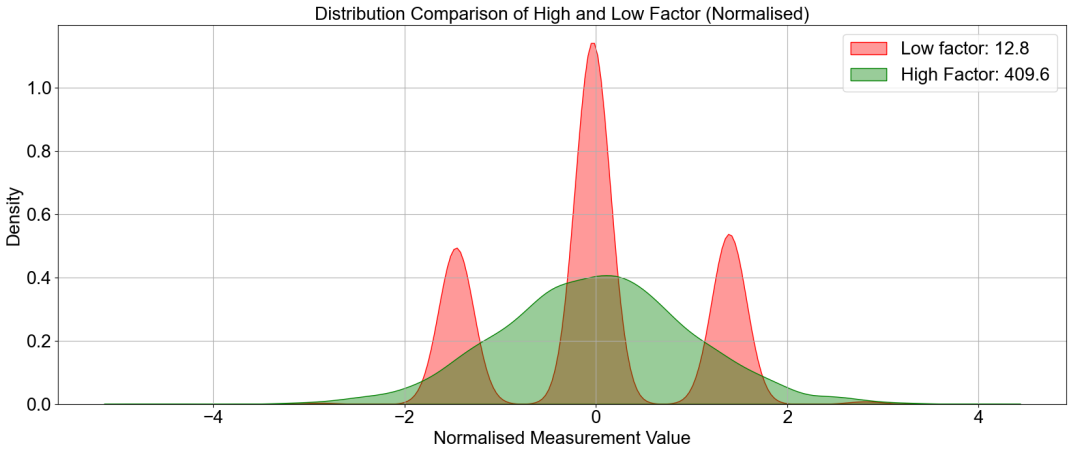

Figure 2.31: Low Factor versus High Factor Distribution graph

For very low FDZP Factors, the measurements deviate from a normal probability density function, resulting in unreliable precision calculations. This phenomenon is indicated by the dashed line in Figures 2.29 and 2.30. In such cases, the probability density function resembles the red distribution shown in Fig. 2.31, in contrast to the well-formed, approximately normal distribution observed at higher Factor values.

Lastly, it is worth noting that increasing the FDZP factor places a significant computational burden on the system, resulting in a pronounced decline in the sensor's cycle frequency. As the Factor increases, its computational demands rise accordingly, eventually slowing the program to a near standstill. This performance degradation occurs earlier for CamR than for CamL. In addition to potential minor manufacturing variances and the added wear experienced by CamR due to extended use, a plausible explanation for this discrepancy is thermal throttling. When a RPi, particularly models such as the Pi 4 or Pi 5, experiences elevated temperatures, it automatically reduces the operating frequency of the CPU and/or GPU. This behaviour helps prevent overheating. While this protective mechanism safeguards the hardware, it inevitably results in reduced performance, as seen in Fig. 2.32 [43]. The root cause of the overheating was later identified as an error in the wiring of the RPi's cooling fan [44]. This issue was resolved prior to the measurements described in the following chapters.

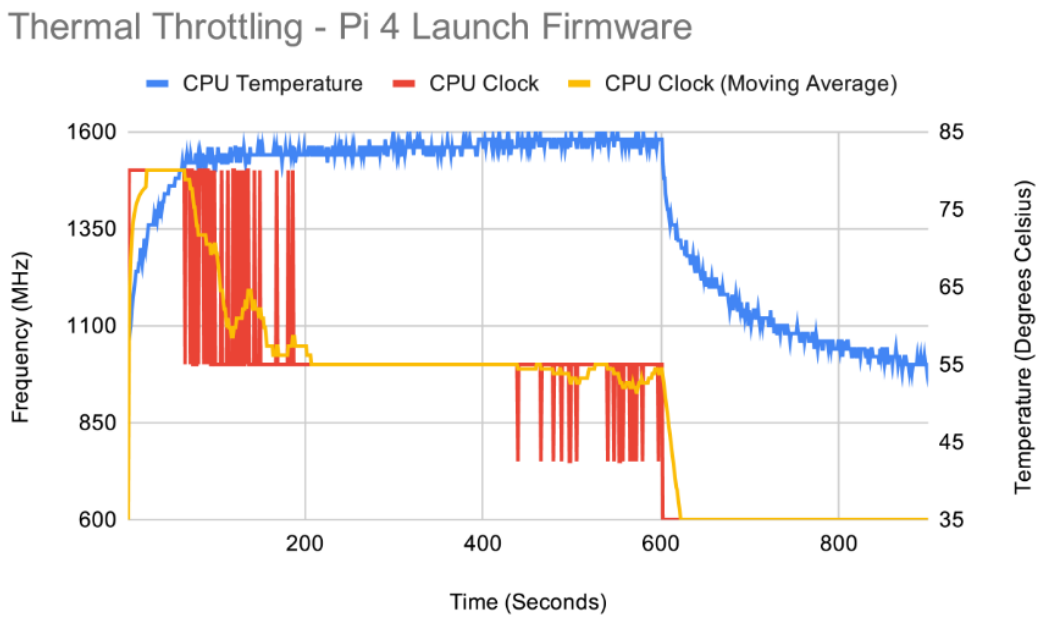

Figure 2.32: Raspberry Pi version 4 Thermal Throttling CPU clock slowdown protection (Raspberry Pi Forum)

Taking all of these considerations into account, the optimal working point for the edge detection sensors is indicated in green in Figures 2.29 and 2.30. This region lies just before the notable decline in cycle time, while still offering a level of precision comparable to that achieved with significantly higher FDZP Factor values. For all subsequent testing, the FDZP Factor will be set to a value of 409.6. This corresponds to a subpixel goal of 8192, or 2^{13}, for a ROI consisting of 20 pixels.

The same subpixel goal is applied to both cameras in order to minimise the variance in cycle time between them, thereby aiming to reduce the synchronisation error previously discussed in 2.6.3. Although cycle time does not limit the FDZP edge detections performed during the preprocessing stage, the processing stage retains the selected subpixel goal. This is done for consistency. The structure of the overall measurement justifies the decision, as it comprises the position of the material edge, determined during processing, and the position of the roller edge, determined during preprocessing. Since the sensor derives the final value it transmits from the subtraction of these two positions from a fixed reference of 670 mm, increasing the precision of the preprocessing result provides no added benefit. While technically feasible, doing so would not contribute to improved measurement precision in the final output.

2.7.2 Comparison with ultrasonic sensor

To assess the viability of the edge detection sensor, and, by extension, the stretch measurement sensor, a comparison was conducted. The reference used was an industrially relevant edge detection system. The WPM is equipped with an ultrasonic sensor [45] specifically designed to perform precise measurements of the material edge within the machine. Its primary function is to ensure accurate winding of the processed material onto a bobbin, as well as to provide edge position data for aligning the pivot table situated at the midpoint of the WPM.

For this comparison, both the RPi-based and ultrasonic sensors were configured to transmit data to the WPM PLC server using the OPC UA protocol over TCP/IP. This data was then retrieved in MATLAB and plotted for direct comparison.

In accordance with the ISO 5725-1 standard, sensor precision was evaluated through a steady-state test. During this test, data from each sensor was recorded over a five-minute interval while the material in the WPM remained stationary. The standard deviation of the recorded values serves as a quantitative measure of the sensor's precision under stable conditions.

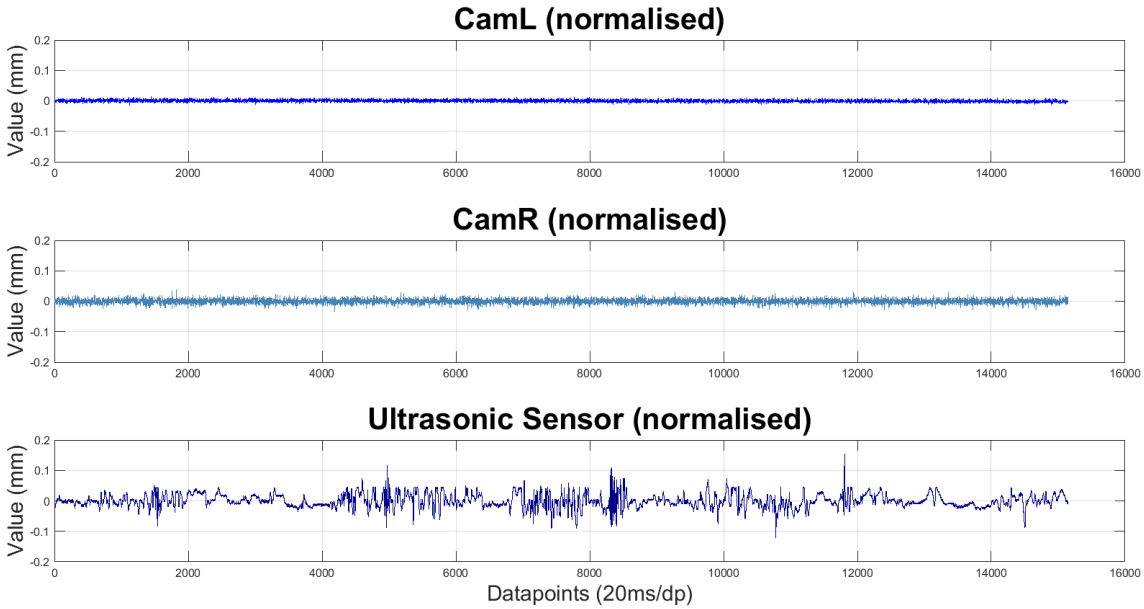

Figure 2.33: Stationary measurement comparison between RPi and ultrasonic edge sensors

Figure 2.33 presents the results of the steady-state tests conducted for both RPi cam-

eras and the ultrasonic sensor. The corresponding measurement statistics for each sensor are summarised in Table. 2.1.

Table 2.1: Summary of measurement statistics for each sensor

Sensor	SD (µm)	Max (µm)	Min (µm)	Range (µm)
CamL	3.978	13.547	-13.186	26.733
CamR	7.417	39.497	-34.729	74.226
Ultrasonic	22.349	154.074	-120.926	275.000

As evident from Fig. 2.33, the measurements obtained from the ultrasonic sensor exhibit more pronounced fluctuations than those from the RPi sensors. Moreover, while the fluctuations in the RPi-based measurements are centred around 0 millimetres, the ultrasonic sensor data displays regions with a consistent offset. This occurs despite both measurement sets being expected to oscillate around a zero reference. These observations indicate that the RPi edge sensors provide more precise and stable measurements compared to the ultrasonic sensor.

The data presented in Table 2.1 further support this conclusion: the RPi sensors demonstrate superior precision, a narrower measurement range, fewer outliers, and smaller peak deviations. It is worth noting that the right camera (CamR) was positioned at a distance of 60 centimetres from the material edge, while the left camera (CamL) was placed closer, at 54 centimetres. This difference in placement accounts for the slight difference in precision observed between the two cameras.

Note that, although the RPi sensors obtain more precise measurements than the ultrasonic sensor, generating each edge measurement takes significantly longer. In terms of response time, the ultrasonic sensor operates at an approximate cycle frequency of 167 Hz, in contrast to the 30 Hz achieved by the RPi sensors. Even with a potential hardware upgrade to the latest RPi model, the RPi 5, the cycle frequency would only increase to approximately 60 Hz. Thus, while the RPi sensors offer superior precision, this improvement comes at the cost of reduced measurement frequency and slower response time. This is a trade-off that must be considered when evaluating their suitability for time-sensitive industrial applications.

It should also be noted that variations in material type can influence the precision of the RPi-based sensor. Figure 2.34 displays three different materials tested within the WPM. The cardboard and thin photo mesh yielded precisions ranging between 2 and 10

micrometres, whereas the kerdi waterproofing membrane material demonstrated lower precision, typically between 10 and 20 micrometres. This difference can be attributed to the quality of the material edges. Fabrics and woven materials tend to have less well-defined edges, which can diminish the effectiveness of edge detection algorithms. In contrast, materials such as paper and carton generally possess sharp, clearly delineated edges, making them more suitable for high-precision edge detection.

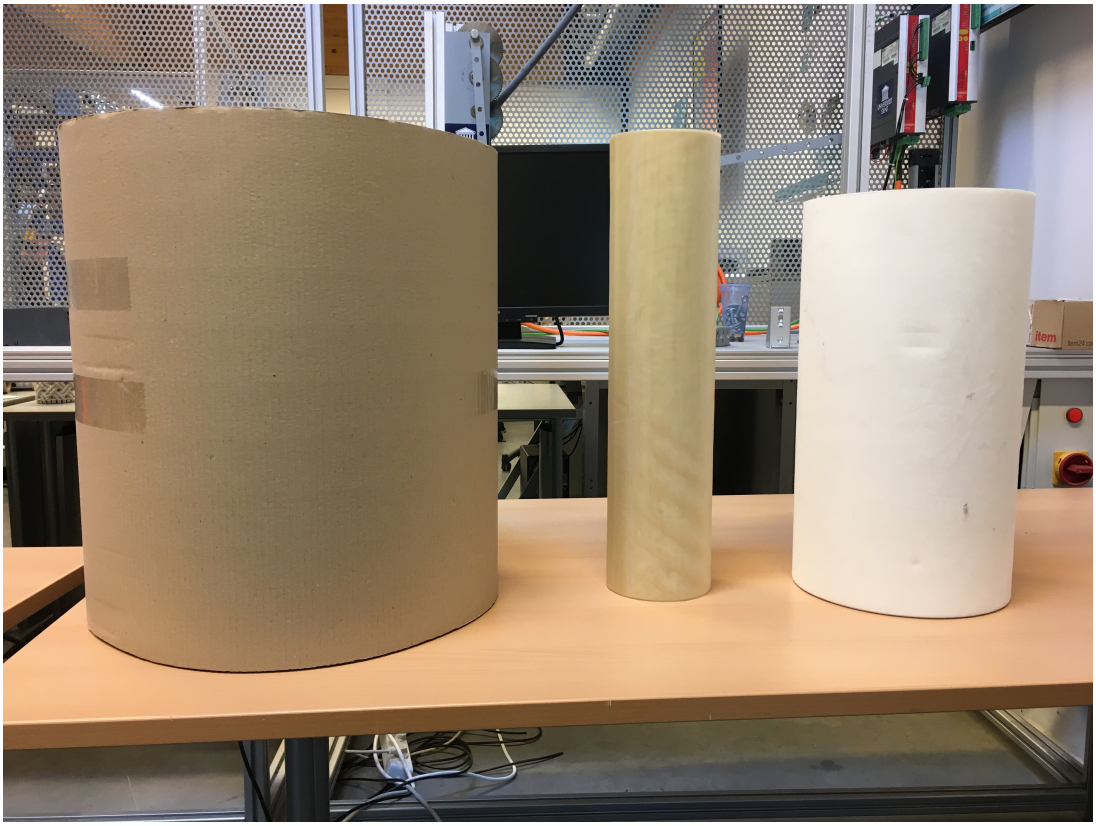

Figure 2.34: Different materials used inside the WPM: Left: cardboard, Middle: thin photo mesh, Right: kerdi waterproofing membrane

An additional observation regarding the thin photo mesh is that it falls below the detection threshold of the ultrasonic sensor, rendering it effectively undetectable by the sensor. In contrast, the RPi-based sensor is capable of reliably detecting a well-defined edge on the same material under standard lighting conditions. This further highlights the advantages of low-cost vision-based systems in edge detection and stretch measurement, particularly in scenarios involving lightweight or semi-transparent materials where traditional sensors may fail.

2.7.3 Width measurement precision

By subtracting the measurements from both cameras from 670 mm, the known length of the roller, a steady-state width measurement can be obtained. This normalised measurement is shown in Fig. 2.35. The expected precision of the stretch sensor, derived from the standard deviations of the individual edge sensors, is calculated to be 8.06 micrometres, as shown in Eq. 2.7 [46].

$$\sigma_{\text{width}} = \sqrt{\sigma_1^2 + \sigma_2^2} \tag{2.7}$$

The observed standard deviation of the width measurement, 8.43 micrometres, aligns closely with this theoretical value, thereby validating the precision estimation.

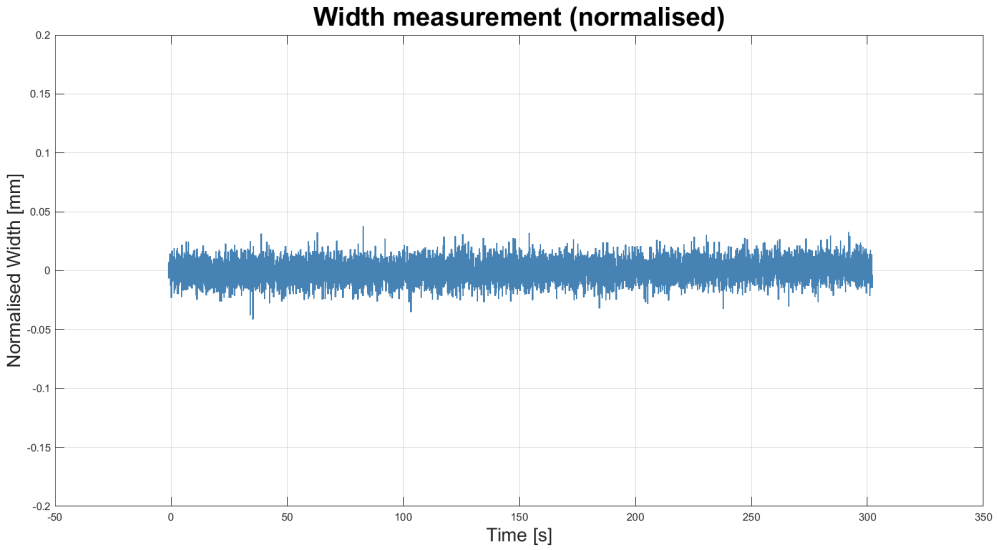

Figure 2.35: Normalised static width measurement data used to calculate sensor precision

Taking into account potential variations in material edge quality and the distance between the camera and the material, the final precision range of the stretch sensor can be defined as lying between 2.8 and 28.28 micrometres.

2.8 Summary of development phase

In this chapter, the development of a low-cost, vision-based stretch sensor employing RPi version 4B units and camera modules was presented. To address the inherent limitations of affordable hardware, several strategies were implemented. These include

FDZP to enhance subpixel edge detection precision and parallel processing techniques to optimise the achievable measurement frequency. A robust pixel-to-millimetre conversion method was developed based on a calibrated roller marker, with intrinsic camera calibration further mitigating lens-induced distortions. Preliminary validation suggests that the RPi-based system is capable of achieving a precision ranging between 2.8 and 28.3 micrometres. This offers measurement stability that is at least comparable to, and in certain aspects potentially exceed, that of an industrial ultrasonic edge detection sensor. In terms of the initial objectives, the goal of attaining a precision within the range of 20 to 100 micrometres was clearly met. However, achieving the targeted measurement frequency of 50 to 500 Hz proved to be beyond the capabilities of the current hardware. Nevertheless, it is expected that modest upgrades, such as the use of more recent RPi models or alternative low-cost processing units, could significantly improve the attainable cycle frequency. Notwithstanding its lower measurement frequency relative to high-end industrial solutions, the proposed sensor presents a viable alternative, particularly in applications where cost efficiency, precision, and adaptability to different material types are of primary importance. While further testing under dynamic conditions and across broader material categories is warranted, the findings indicate that low-cost vision-based stretch sensing may hold significant potential for integration into web processing machinery.

3

Practical sensor evaluation

Having completed the development of the RPi-based stretch sensor, the next step is to evaluate its performance under realistic industrial conditions. This chapter investigates the sensor's behaviour in the dynamic environment of a WPM, where material moves continuously and experiences variable tension. The aim is to determine whether the sensor delivers precise, stable, and interpretable measurements of stretch and deformation, essential requirements for integration into a real-time control system. The first step involves comparing the sensor's output with that of a conventional ultrasonic setup to benchmark precision, signal quality, and responsiveness. Next, a series of tensile tests on moving material characterise the sensor's ability to detect both elastic and plastic deformation under load. Time-domain and frequency-domain analyses together build a detailed picture of the sensor's practical capabilities and limitations: precision, stability, interpretability, and responsiveness. This evaluation lays the foundation for its deployment in a closed-loop control scenario in the following chapter.

3.1 Comparing sensor outputs on moving material

To facilitate comparison and evaluate potential internal or external sources of measurement error, the stretch sensor's output is benchmarked against that of an alternative, makeshift sensor setup. This reference sensor consists of two ultrasonic edge detection sensors, previously introduced in Chapter 2. They are positioned on opposite sides of the material, at its edges. By summing the distance readings from both sensors and adding the known roller length between them, the system obtains an approximate estimate of the material's width.

Figures 3.1 and 3.2 present the width measurements obtained from the RPi-based stretch sensor and the ultrasonic reference sensor respectively. The system recorded

these measurements simultaneously on moving material inside the WPM. A roll of white kerdi material was used, operated at a constant speed setpoint of 10 m/min and a tension setpoint of 80 Newton. A phase shift of approximately two seconds appears between the two signals. This shift corresponds to the time taken by the material to travel from one sensor to the other within the machine.

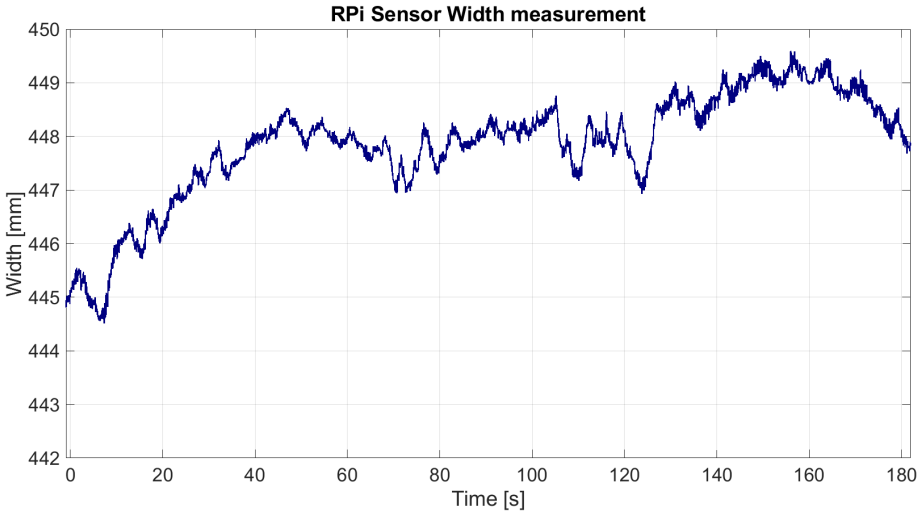

Figure 3.1: RPi sensor width measurement on moving material

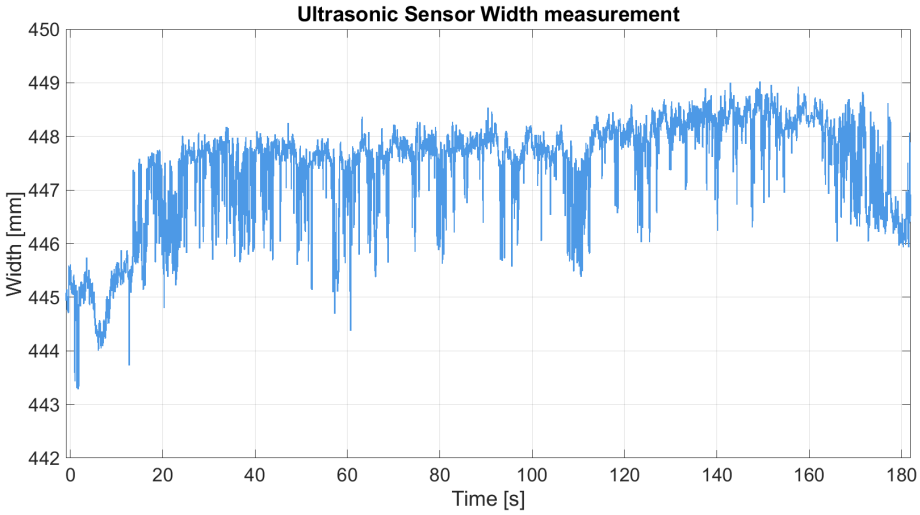

Figure 3.2: Ultrasonic sensor width measurement on moving material

3.1.1 Correlation between vision and ultrasonic data

One concern with measuring the material width at a roller within the WPM is that localised deformation may occur. These deformations cause variations in the measured

width because the measurement location is fixed at the roller, rather than at a point where the material is freely suspended. To assess whether the RPi sensor captures the same deformation patterns as the ultrasonic sensor, the system computes the Pearson Correlation Coefficient.

The Pearson Correlation Coefficient quantifies the strength and direction of the linear relationship between two variables: it measures how closely their values co-vary around their respective means [47][48]. Its value ranges from +1, indicating perfect positive correlation, to –1, indicating perfect negative correlation, with 0 denoting the absence of a linear relationship. In this context, the coefficient assesses whether both sensors capture comparable variations in material width over time. Mathematically, the Pearson correlation coefficient r is defined via Eq. 3.1.

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3.1}$$

where x_i and y_i represent the individual measurements from the RPi camera-based width estimates and the ultrasonic sensor-based width estimates, respectively, and $\bar{x}$ and $\bar{y}$ are their corresponding sample means. The total number of time-aligned samples is denoted by n.

To evaluate the statistical significance of the observed correlation, a two-tailed hypothesis test is performed using the t-statistic, as defined by Eq. 3.2.

$$t = r \cdot \sqrt{\frac{n-2}{1-r^2}} \tag{3.2}$$

This t-value follows a Student's t-distribution with $n-2$ degrees of freedom, from which a p-value is computed. A small p-value (typically < 0.05) indicates that the correlation is unlikely to have occurred by random chance, suggesting a statistically significant linear relationship between the two sensor signals.

High-frequency noise in the raw sensor data made it necessary to smooth both signals using a moving average filter to suppress short-term fluctuations. Figure 3.3 shows the resulting smoothed signals. To improve visual distinction, the system offset the ultrasonic sensor signal by –2 millimetres.

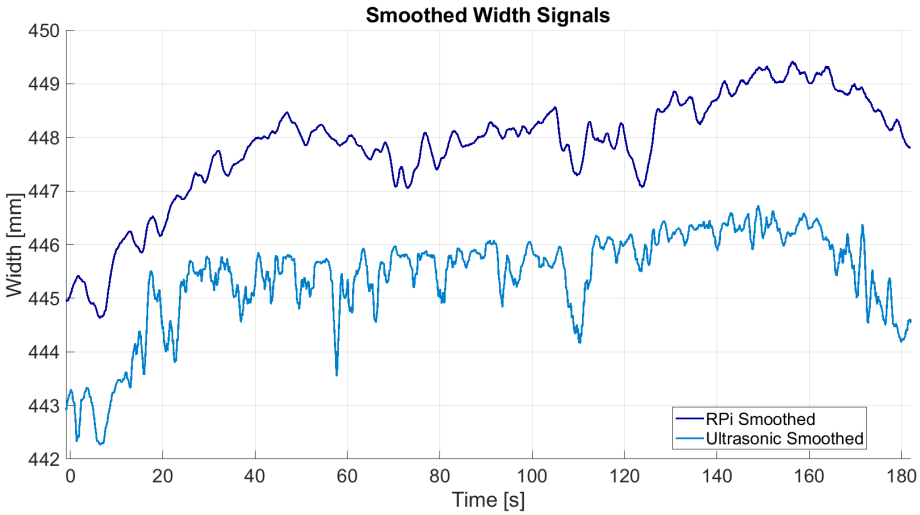

Figure 3.3: Smoothed width signals (Ultrasonic offset by -2 mm) used for correlation comparison

This filtering ensures that the correlation calculation captures the underlying deformation pattern, rather than being skewed by uncorrelated sensor noise. Without smoothing, the correlation would likely underestimate the true similarity between the signals, as discrepancies in short-term fluctuations could obscure the broader trends shared by both sensors.

The Pearson correlation coefficient calculated for the two signals was $r = 0.8319$, indicating a strong positive linear relationship. This result suggests that the RPi and ultrasonic sensors both respond to the same global deformation pattern over time. However, this method primarily captures broad trends and remains relatively insensitive to small-scale or localised fluctuations. As a result, while the correlation coefficient provides a useful initial consistency check, it should be interpreted with caution in contexts where high precision is required.

Given the relatively high noise levels and limited resolution of the ultrasonic sensors, it is likely that they cannot reliably detect finer deformation features. Therefore, although a high correlation value is encouraging, it does not imply complete agreement between the two signals at a detailed level. Moreover, any DC offset between the measurements is difficult to quantify. This is because the ultrasonic sensor's output depends critically on the accurate measurement of the physical distance between the sensors, determined in this case using a calliper.

In addition to the correlation coefficient, the p-value assesses the statistical significance

of the observed relationship. It addresses the following question: if there were no actual correlation between the two signals, what is the probability of observing a correlation of this magnitude purely by chance? A p-value of 0.00 (i.e., $p < 0.001$) indicates that such a result is extremely unlikely to occur randomly, thereby confirming that the observed relationship is statistically significant. In this case, the high correlation is not only strong but also demonstrates a high degree of reliability.

3.1.2 Precision and signal quality

Standard Deviation on moving material

The standard deviation measures how much a signal varies around its mean and is commonly used as an indicator of its stability and precision. In this context, it captures both the natural changes in the material's width and any extra fluctuations introduced by the sensors.

In this case, both the RPi and ultrasonic sensors report nearly identical standard deviations (approximately 1.02 mm). This might misleadingly suggest that both sensors exhibit comparable levels of stability and precision. However, this interpretation overlooks a key nuance. The ultrasonic sensor displays noticeably more high-frequency noise, while the RPi sensor produces a much smoother signal. The core issue is that standard deviation does not differentiate between two types of variation: slow, meaningful changes such as actual width deformation, and rapid fluctuations caused by sensor noise. Both contribute equally to the final standard deviation. As a result, a noisy signal with minor high-frequency jitter can yield the same standard deviation as a clean signal that faithfully tracks material deformation. This makes standard deviation a potentially misleading metric for evaluating sensor performance in dynamic scenarios, such as moving material, in contrast to the stationary conditions considered in Chapter 2.

Signal to Noise Ratio (SNR)

The signal-to-noise ratio (SNR) was employed to assess the clarity of the deformation signal relative to high-frequency noise for each sensor. This metric was calculated in the frequency domain by dividing the total energy within a designated signal band (0.01–2.0 Hz) by the energy in a noise band. The signal band was manually selected to correspond with the range of physically meaningful width variations induced by material motion. The noise band was defined as all frequencies above 80% of the Nyquist limit.

This method effectively isolates the energy associated with actual deformation from that arising due to high-frequency sensor instability.

Using this frequency band definition, the RPi sensor achieved an SNR of 14.74, whereas the ultrasonic sensor yielded a considerably lower SNR of 5.47. This substantial difference clearly indicates that the RPi sensor captures deformation signals with significantly greater clarity and reduced high-frequency interference. In contrast, the ultrasonic sensor demonstrates a higher degree of short-term fluctuations and measurement instability, as reflected in its lower SNR.

It is important to note that earlier SNR evaluations, based on narrower or less representative frequency bands (e.g., 0.1–1.0 Hz), occasionally underestimated this difference or were distorted by structured noise present within the signal band. By accurately aligning the selected frequency range with the physical behaviour of the system, the SNR calculation becomes a more reliable indicator of sensor performance. These findings corroborate what is already evident in the time-domain representations: the RPi sensor delivers a cleaner, more precise measurement of material width, with substantially less pronounced fluctuations.

High-Frequency Energy

The High-Frequency Energy metric quantifies the proportion of spectral energy present above 2 Hz, a threshold beyond which no significant deformation of the material is expected to occur. Given that deformation resulting from material motion predominantly occurs below this frequency, any spectral content above 2 Hz is attributed to extraneous noise sources. Examples of this are: sensor jitter, electronic interference, or optical tracking artefacts. To obtain this metric, the spectral magnitudes corresponding to frequencies above 2 Hz were integrated. This metric is dimensionless and not physically interpretable in absolute terms; it is used solely for relative comparisons.

In the present analysis, the RPi sensor exhibited a High-Frequency Energy of 19,411, whereas the ultrasonic sensor reached a markedly higher value of 99,119, more than five times greater. This stark contrast confirms that the ultrasonic sensor signal contains a substantial amount of undesirable high-frequency fluctuations.

As this metric specifically targets the portion of the frequency spectrum that should ideally remain quiet in a stable measurement, it offers a clear and interpretable indication of signal contamination. It is particularly valuable for assessing sensor performance under dynamic conditions, where measurement integrity is critical.

Noise Standard Deviation

The noise standard deviation was calculated by first subtracting the moving average, as seen in Fig. 3.3, from each raw signal to remove slow-varying components. This subtraction effectively acts as a high-pass filter, isolating the high-frequency residual, otherwise seen as the local noise around the main deformation trend. The standard deviation of this residual then quantifies the sensor's short-term instability. This can be seen in Figures 3.4 and 3.5 for RPi and ultrasonic sensors respectively.

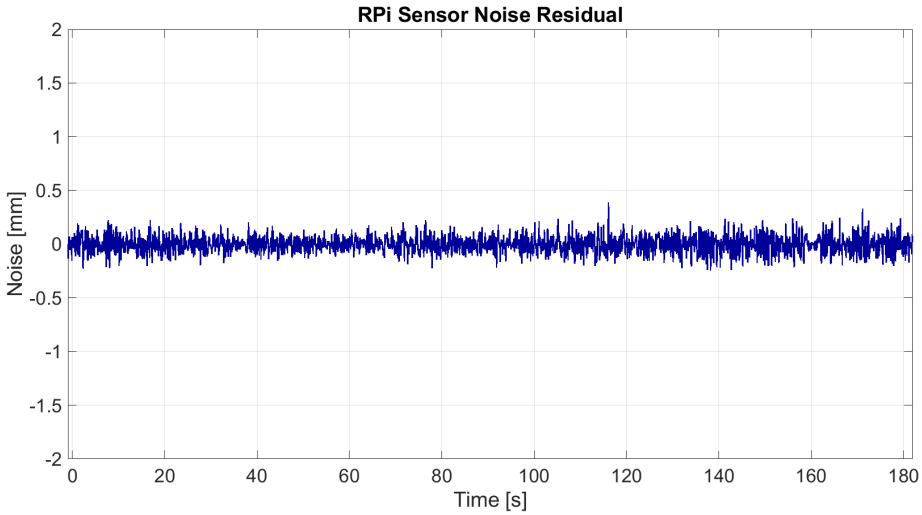

Figure 3.4: RPi sensor noise residual used to calculate noise SD

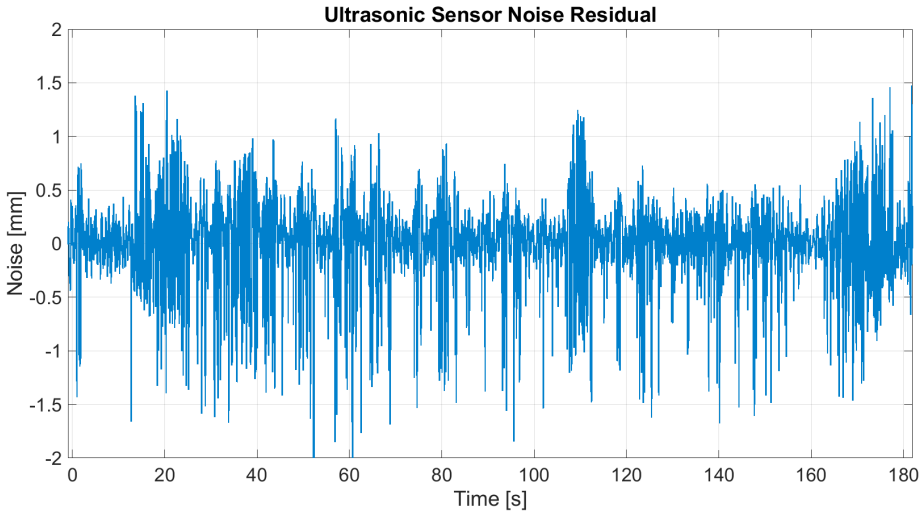

Figure 3.5: Ultrasonic sensor noise residual used to calculate noise SD

The RPi sensor produced a noise standard deviation of 0.075 mm, while the ultrasonic sensor exhibited 0.412 mm, over five times greater. This confirms that the ultrasonic sensor introduces significantly more high-frequency noise into the measurement. Unlike total standard deviation, which can be inflated by real material deformation, this metric focuses purely on the fast, undesired fluctuations. As a result, it provides one of the most reliable and interpretable measures of sensor precision in the context of a moving material. Table 3.1 summarises the comparison metrics.

Table 3.1: Comparison of signal quality metrics between Ultrasonic and RPi sensors

Metric	**Ultrasonic Sensor**	**RPi Sensor**
SD [mm]	1.02	1.02
SNR [–]	5.47	14.74
High-Frequency Energy [a.u.]	99,119	19,411
Noise SD [mm]	0.412	0.075

Comparison conclusion

Both sensors were found to track the same large-scale deformation trends in the material, confirming consistent response across measurement positions. However, more detailed frequency and noise analyses reveal that the RPi-based sensor outperforms the ultrasonic sensor in terms of precision, stability, and signal quality. The RPi sensor exhibits substantially lower high-frequency energy and a much smaller noise standard deviation, alongside a higher signal-to-noise ratio. In addition to its superior signal characteristics, the RPi setup also presents a clear cost advantage, being approximately four times cheaper than the ultrasonic system, while offering a wider and more flexible working range. These results indicate that while the ultrasonic sensor is suitable as a basic reference device, the RPi stretch sensor provides a cleaner, more robust, and more economical solution, making it the preferred choice for precision industrial applications.

3.2 WPM-based stretch characterisation of material

To evaluate the viability and performance of the RPi stretch sensor, a force–extension curve for the transverse stretching of material within the WPM was experimentally measured and plotted.

3.2.1 Description of test material

For this test, a roll of kerdi membrane was utilised. Kerdi is a flexible, sheet-applied waterproofing membrane primarily designed for installation beneath tile coverings in wet environments [49]. It comprises a thin, impermeable layer of polyethylene (PE), a chemically stable and waterproof thermoplastic polymer [50], laminated on both sides with a non-woven polypropylene fleece [51]. This dual-layer construction promotes mechanical anchorage to thin-set mortar and imparts a distinctive web-like surface appearance. Although it should be noted that the fleece is a nonwoven structure rather than a true woven fabric.

Mechanically, kerdi exhibits the typical behaviour of polyethylene-based composites, characterised by a low elastic modulus, high ductility prior to rupture, and good flexibility, enabling it to conform to irregular surfaces. As with most polyolefin materials, it shows relatively low resistance to cyclic fatigue and creep under sustained loading.

Within the WPM, kerdi exhibits clear and measurable deformation under tension without a significant risk of tearing or rupture, making it a suitable and safe material for testing the measurement capabilities of the RPi stretch sensor. Its mechanical properties allow for a dynamic, multi-layered evaluation of sensor performance. Its robustness ensures safe operation throughout testing. This, in turn, reinforces the sensor's relevance for potential industrial applications.

3.2.2 Setup and test conditions

Controlled tensile tests were performed to investigate the transverse mechanical response of kerdi. The membrane was continuously fed through the WPM while subjected to incrementally increasing tension forces. Stretch measurements were recorded using the RPi sensor during motion. Following each force application, the tension was reduced to a low reference value of 20 Newtons to measure any residual deformation, thereby separating recoverable (elastic) deformation from permanent (plastic) deformation. The reference measurement serves as the new baseline for subsequent load steps.

Although nominal force increments were kept constant, progressive material degradation after each load cycle resulted in a slight reduction in resistance, causing the effective deformation steps to become increasingly non-uniform. Ideally, an identical fresh sample would have been used for each load step to eliminate this effect, however, time and budget constraints precluded this approach.

The kerdi roll used has an initial width of approximately 450 mm, consistent with typical commercial dimensions.

3.2.3 Material characterisation test results and observations

The results of the measurement campaign are presented in Figures 3.6 and 3.7. Figure 3.6 shows two curves: the light blue curve represents the average material width under constant force during motion, and the dark blue curve shows the residual stretch after unloading to the 20 N reference force. These datasets were combined to construct the force-extension curve shown in Fig. 3.7.

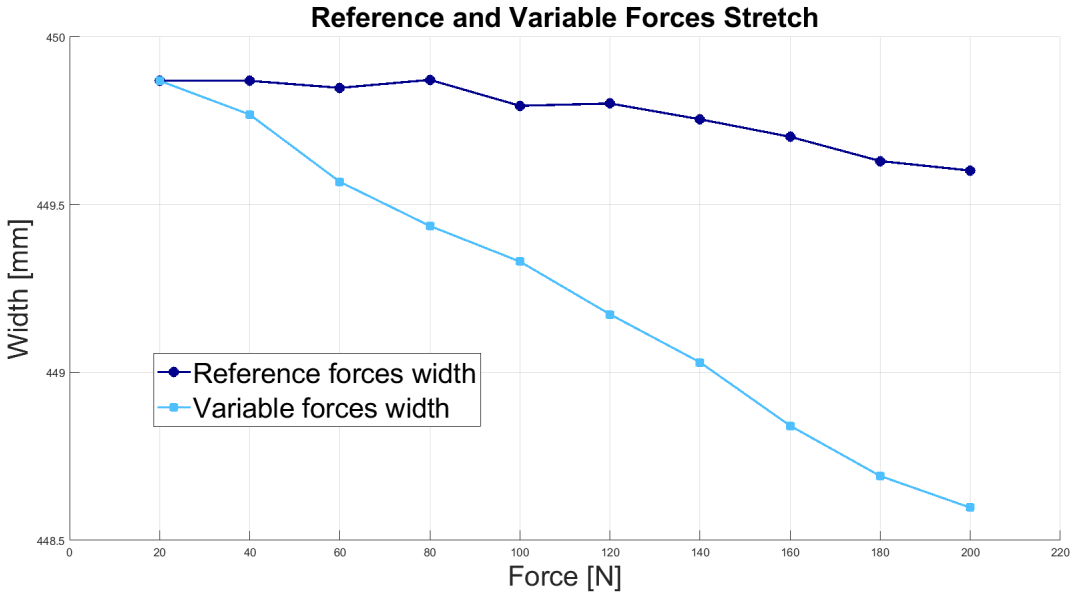

Figure 3.6: Average measured width during force application and after force removal

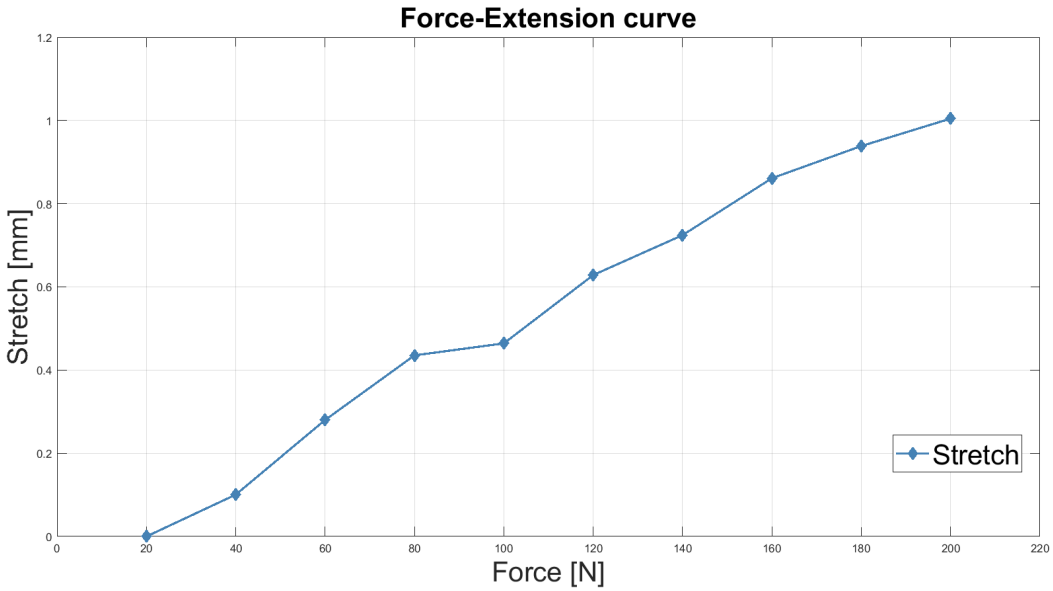

Figure 3.7: Resulting force-extension curve from measurement results

The force-extension data revealed a near-linear relationship between applied force and transverse stretch within the tested range of 20 N to 200 N. At the maximum force, the total observed stretch was approximately 1.0 mm across the 450 mm width. This outcome indicates a relatively low elastic stiffness, consistent with the known mechanical behaviour of polyethylene-based laminates [49]. No discontinuities, abrupt slope changes, or signs of rupture were observed, suggesting that the material underwent smooth elastic and early plastic deformation without failure.

The residual measurements taken after unloading confirmed the gradual accumulation of plastic deformation. After each force step, the material's width at the reference force had decreased slightly relative to its original value. The progressive widening between the loaded and reference curves indicated controlled plastic flow, consistent with a ductile material response.

3.2.4 Interpreting dynamic behaviour

The force-extension data collected by the RPi sensor demonstrated a plausible measured relationship between force and transverse stretch for the kerdi material. This aligns closely with known behaviours of polyethylene-based composites. This indicates that the sensor is correctly capturing the physical deformation of the membrane without introducing significant artefacts.

The comparison between loaded and reference measurements further validated the sensor's performance. The test sequence clearly detected gradual plastic deformation. The consistent widening between the curves as force increased reflected good sensitivity and stability under dynamic web-processing conditions. Minor variations observed at low force levels were within reasonable expectations, given environmental factors and machine vibrations.

Nevertheless, some caution is warranted when interpreting the absolute values of stretch. As the same section of material was subjected to repeated loading, progressive softening resulted in slightly greater deformations at each nominal force increment. Although this effect was anticipated, it introduces some uncertainty into the quantitative interpretation of the force-extension relationship. Ideally, a more rigorous validation would involve testing a fresh, identical material roll at each load step; however, budgetary and time constraints made this impractical within the scope of this master's thesis.

In summary, the RPi stretch sensor demonstrated good feasibility and performance under industrially relevant conditions, providing reliable and coherent measurements of material deformation suitable for web-processing applications.

3.3 Summary of measurement phase

This chapter assessed the practical performance of the developed low-cost RPi stretch sensor under dynamic industrial conditions. Through a comparative study with ultrasonic reference sensors, the RPi sensor demonstrated its ability to reliably track large-scale material deformation trends, while offering significantly enhanced signal clarity, stability, and noise performance. Furthermore, tensile testing on kerdi membrane confirmed that the sensor accurately captured both elastic and plastic material responses during web processing. Although the process identified certain limitations related to progressive material degradation, the overall findings suggest that the RPi stretch sensor constitutes a robust and cost-effective solution for real-time stretch measurement, with potential for industrial integration.

4

Quality control inside the WPM

With the stretch sensor's performance now established in controlled conditions, the next step is to evaluate its real-world applicability within an industrial context. This chapter explores the integration of the RPi-based sensor into a functioning WPM. This with the goal of demonstrating that the sensor is not only accurate, but also pragmatic and deployable in dynamic production environments. Specifically, the sensor is used as the foundation for a closed-loop feedback system that regulates material tension in real time, a critical factor influencing web quality. Through the development and tuning of a layered control strategy, the system responds to width measurements from the RPi sensor to dynamically adapt tension and mitigate undesirable deformations. Along the way, the chapter examines both the technical challenges of implementing responsive control in an industrial machine. This includes sensor resolution limits, actuation delay, and filter design, and the measurable impact of this control on material behaviour.

4.1 Overview of tension control inside the WPM

Before developing a control system capable of influencing the material tension, it is essential to first understand how tension is applied within the WPM. Figure 4.1 provides an overview of the key components involved in the tensioning process, as well as the placement of the RPi stretch sensor within the machine.

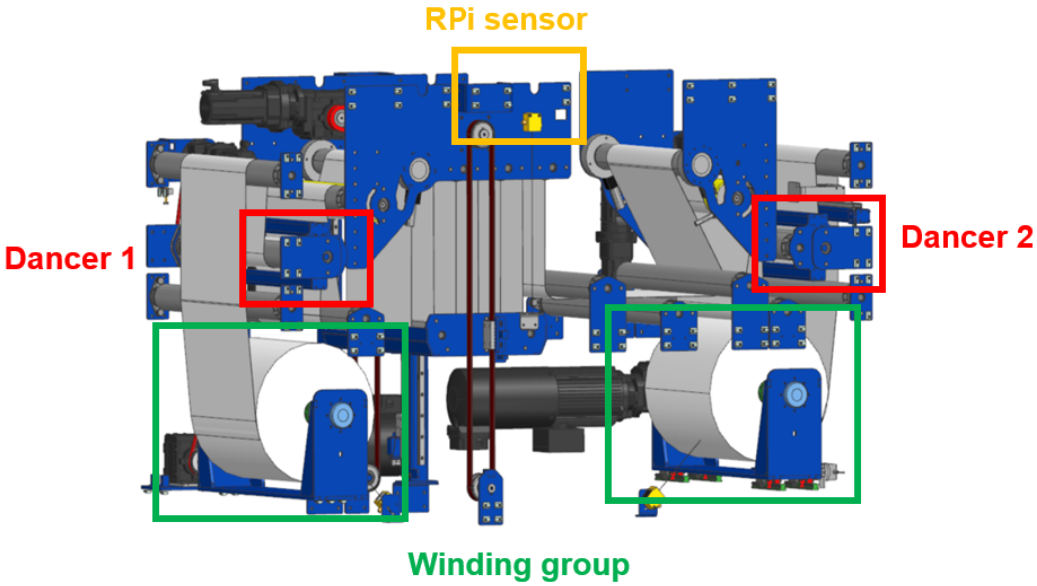

Figure 4.1: WPM Model with Highlighted Components

As material is unwound from one winding roller and transferred to another, typically in one of two possible directions, it passes through various components of the WPM. Most rollers in the system can be broadly classified into two categories: *traction rollers* and *free rollers*.

Free rollers are not actively driven by a motor. Instead, they rotate solely due to the movement of the material. They begin to turn only when the frictional force between the web and the roller surface becomes large enough to overcome both the roller's inertia and the static friction in its bearings. This interaction introduces a tension differential across the roller due to energy loss from friction. The relationship between the entry and exit tension around such a roller can be described by the capstan equation [52], which quantifies how friction and wrap angle affect tension, shown in Eq. 4.1.

$$T_2 = T_1 \, e^{\mu\theta} \tag{4.1}$$

In this equation, T_1 is the tension on the entry side of the roller (before contact). T_2 is the tension on the exit side (after contact). μ is the coefficient of friction between the web and the roller surface. And θ is the wrap angle in radians, representing how much of the roller's circumference the material is in contact with.

This formula reveals that even passive elements like free rollers can induce tension differences, especially when the web wraps the roller over a large angle or when friction is high.

In contrast, traction rollers are powered by motors and directly control the speed of the web. These rollers are strategically placed throughout the WPM to actively regulate tension. Their goal is to mitigate any undesirable gradients that may arise between different sections of the machine. By matching or adjusting the surface speed of the roller to the web speed, traction rollers help ensure stable tension throughout the WPM. They are essential for decoupling tension zones, especially during speed changes or start-stop events.

These components define the baseline tension experienced by the material as it progresses through the WPM. However, this inherent tension is relatively modest compared to the tensions that are actively imposed. These can be imposed either manually by the operator or dynamically as part of adaptive, quality-driven control strategies. To enable such controlled tensioning, the WPM employs components such as the dancer units in combination with the winding group.

The dancers, illustrated in Fig. 4.2, consist of two free rollers mounted on a central, motorised axis. The material is routed under the first roller and over the second. By rotating the dancer, a tensioning force is applied to the material, proportional to the torque generated by the motor. When a new tension setpoint is introduced, the dancer absorbs or releases web length by rotating, thereby increasing or decreasing the tension imposed on the web. It is then the responsibility of the corresponding winding roller to compensate for any slack introduced by this movement, thereby maintaining continuous and stable material flow.

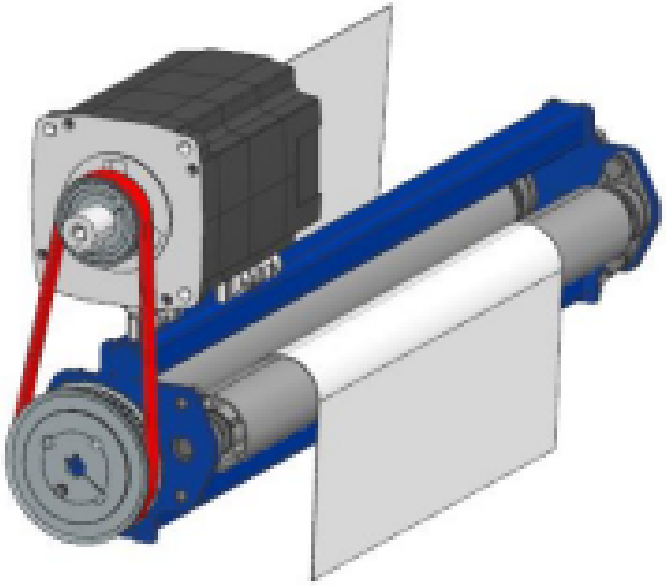

Figure 4.2: Dancer control system with permanent magnet motor and toothed belt transmission

4.2 Tension feedback control objective

Figure 4.3 presents a width measurement dataset captured for use in the force–extension analysis discussed in Chapter 3. Visual inspection of the graph reveals semi-random fluctuations in material width over time. These variations can be attributed partly to deformation induced by the WPM and partly to inherent imperfections or weak points within the material itself, either from manufacturing or from repeated use.

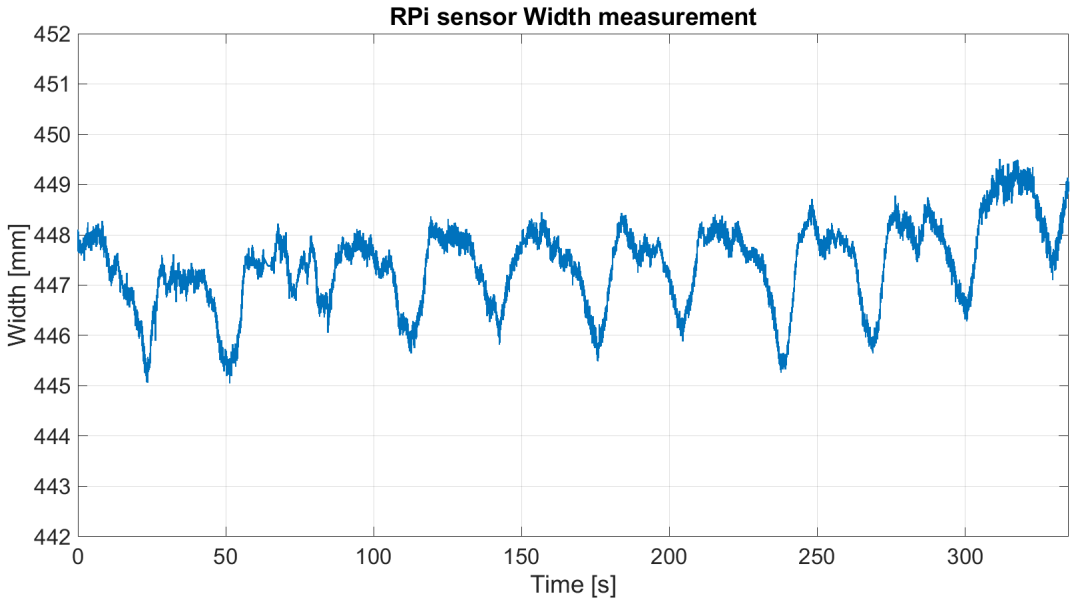

Figure 4.3: RPi sensor width measurement on moving material at low tension after multiple cycles

Ideally, material processed within the WPM should exhibit a consistent width throughout its length. Deformations introduced by the machine should be compensated for, and any irregularities in the material should be minimised or corrected. In theory, a pristine material would yield a width measurement profile resembling a straight line, with a SD equal to the intrinsic precision of the sensor system.

The objective of the proposed tension control system is to dynamically adjust the applied tension based on feedback from the RPi stretch sensor. This with the aim of improving material quality by minimising width fluctuations during processing.

4.3 Tension control logic and implementation

Normally, the dancers operate at a fixed tension setpoint defined by a manually configured variable. Each dancer is governed by a PI controller, which regulates motor torque to maintain the desired tension. The proposed control strategy introduces an additional P controller placed upstream of the existing PI controller, forming a cascaded control structure. This added layer allows the system to dynamically adjust the dancer's tension setpoint based on real-time feedback, rather than relying solely on a static value.

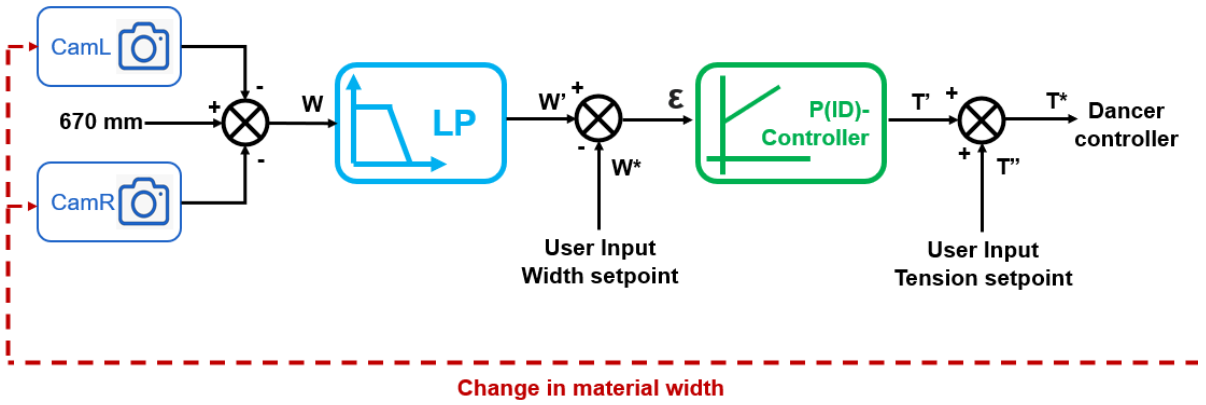

Figure 4.4: Control loop diagram

Figure 4.4 presents a schematic overview of the tension control loop. The process begins by reading the edge measurements from both RPi units via PLC server nodes. These measurements are subtracted from the fixed roller length of 670 mm to determine the current material width. The resulting raw width signal (W) is then passed through a low-pass filter to attenuate sensor noise and suppress high-frequency deformations deemed too small for the control loop to effectively manage. The filtered width (W′) is then compared to a user-defined width setpoint (W*), and the resulting error signal serves as input to the P controller.

The output of this controller (T′) is a tension correction value ranging between -200 and +200 Newton, which can be rescaled if needed. This correction is then added to the user-defined base tension (T″) to form the final tension setpoint (T*) used by the dancer's PI controller. In this way, the system transitions from operating with a fixed tension to a dynamically adjusted setpoint based on real-time width measurements. An important note not shown in the diagram is a limiter implemented on the controller output to ensure that the resulting tension setpoint remains within practical and safe operational bounds. This limiter prevents the system from requesting tension levels that exceed the physical capabilities of the dancer units, or from applying tensions that are too low to maintain proper material handling within the WPM.

4.4 System constraints and practical limits

Before programming of the control loop can commence, it is necessary to first consider the limitations of the system and identify its weakest link, whether this lies in the sen-

sor performance, the positioning of the cameras, or the response speed of the dancer mechanism.

4.4.1 Vision sensor measurement capability limits

When evaluating the limitations of the control loop, it is essential to consider the size and shape of deformations that the stretch sensor is capable of accurately detecting. This determines the level of detail the control system can respond to and ultimately constrains its effectiveness in maintaining consistent material width. The width sensor has a precision of 10 micrometres and samples the moving material at a frequency of 30 Hz. The material moves at a constant speed of 10 m/min, equivalent to 166.7 mm/s. This leads to a spatial sampling interval, calculated using Eq. 4.2:

$$\Delta x = \frac{v}{f_s} = \frac{166.7\,\text{mm/s}}{30\,\text{Hz}} \approx 5.56\,\text{mm}, \tag{4.2}$$

which means the sensor takes one measurement approximately every 5.56 mm along the material.

To assess the sensor's ability to detect wave-like deformations in the material width, periodic deviations can be modelled as sinusoidal variations. According to the Nyquist–Shannon sampling theorem [23], the shortest wavelength that can be detected without aliasing is given by Eq. 4.3:

$$\lambda_{\text{min}} = 2 \cdot \Delta x = 2 \cdot 5.56\,\text{mm} = 11.1\,\text{mm}. \tag{4.3}$$

Therefore, any deformation must have a wavelength of at least 11.1 mm to be resolved unambiguously .

The amplitude of the deformation must also be distinguishable from sensor noise. For single-point deviations, a commonly accepted confidence margin is three times the standard deviation, resulting in a minimum detectable amplitude given by Eq. 4.4

$$A_{\text{min}}^{\text{single}} \approx 3 \cdot 10\,\mu\text{m} = 30\,\mu\text{m}. \tag{4.4}$$

If the deformation spans N samples, averaging can be used to reduce the effect of random noise. The effective noise level decreases with the square root of the number of samples, yielding 4.5:

$$A_{\text{min}}^{\text{avg}} \approx 3 \cdot \frac{10\,\mu\text{m}}{\sqrt{N}}. \tag{4.5}$$

For example, a deformation spread over four samples (approximately 22 mm) would require a minimum amplitude of about 15 micrometres, while one spanning ten samples (roughly 56 mm) could be detected with an amplitude of approximately 9.5 micrometres. For very slow deformations extending over 40 samples (roughly 222 mm), amplitudes as low as 4.7 micrometres can be detected.

Finally, the spatial frequency limit derived from 4.3 translates into a maximum detectable temporal frequency, determined by the product of spatial frequency and material speed as seen in Eq. 4.6:

$$f_{\text{max}} = \frac{1}{\lambda_{\text{min}}} \cdot v = \frac{1}{11.1\,\text{mm}} \cdot 166.7\,\text{mm/s} \approx 15\,\text{Hz}. \tag{4.6}$$

This implies that width deformations occurring up to 15 times per second can be reliably detected as the material moves past the sensor.

In conclusion, a deformation must have a wavelength of at least 11.1 millimetres and an amplitude exceeding 30 micrometres to be detected in a single sample. For longer deformations, amplitudes as low as approximately 5 micrometres can be detected, depending on the number of samples affected. This defines a clear boundary on the size and frequency of imperfections that can be confidently distinguished from sensor noise.

4.4.2 Limits of the dancer system

As illustrated in Fig. 4.1, there is a considerable length of material between the dancer unit and the sensor measurement position. At the standard operating speed of 10 m/min, it was determined that the material requires approximately 3 seconds to travel from the dancers to the sensor in either direction (excluding the accumulator). This introduces a significant delay between the detection of an error and the application of a corrective action.

As a result, with the current sensor placement, only large and slow deformations can be accurately and effectively corrected. This inherent delay represents the primary limitation of the control system and constitutes the most critical bottleneck to its performance.

4.5 Low-pass filter design

One of the key components of the control loop is the filtering of the input signal. The initial approach employed a moving average filter. This filter operates by continuously averaging the most recent X raw input values to smooth out short-term fluctuations and sensor noise. A circular buffer stores the latest values, and each new input replaces the oldest entry. The sum is updated incrementally, and the filtered output is computed by dividing the sum by the number of stored values.

Figure 4.5 illustrates the effect of a small moving average filter. While this configuration introduces a delay of approximately 0.4 seconds, it fails to sufficiently suppress smaller variations in the width signal. As a result, the controller may respond too aggressively, potentially leading to overshoot and the amplification of minor deformations.

By contrast, Fig. 4.6 shows the application of a significantly larger moving average filter. This filter more effectively attenuates small fluctuations, thereby preventing the controller from reacting to uncorrectable disturbances. However, this comes at the cost of increased latency, introducing a time delay of roughly 1 second. Consequently, the overall response time of the control system increases from approximately 3 to 4 seconds. This limits its ability to respond promptly and reducing its effectiveness in mitigating dynamic material deformations.

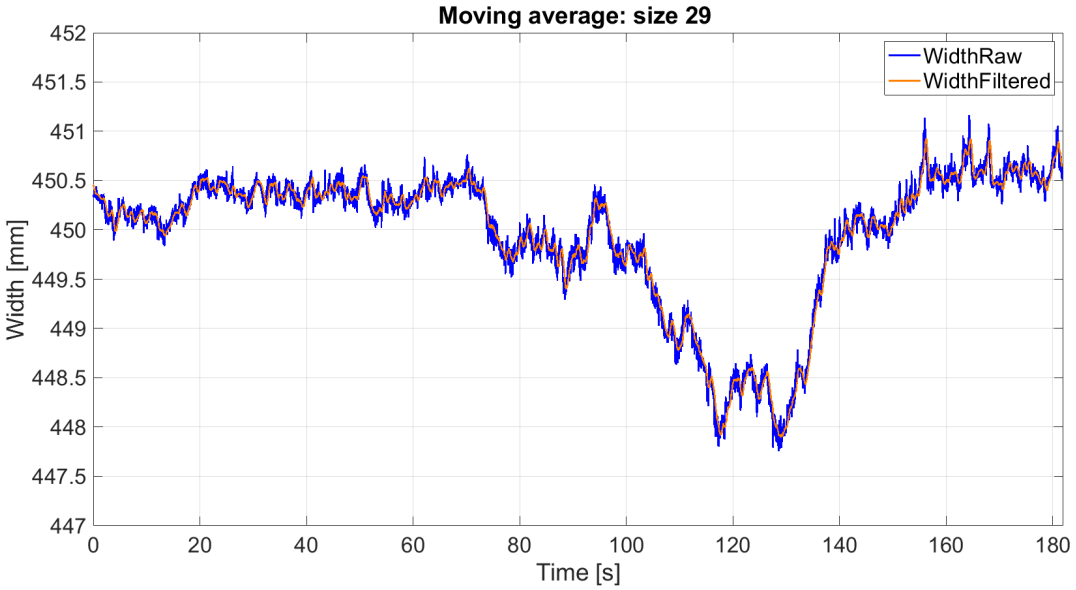

Figure 4.5: Width data with moving average filter applied (size 29)

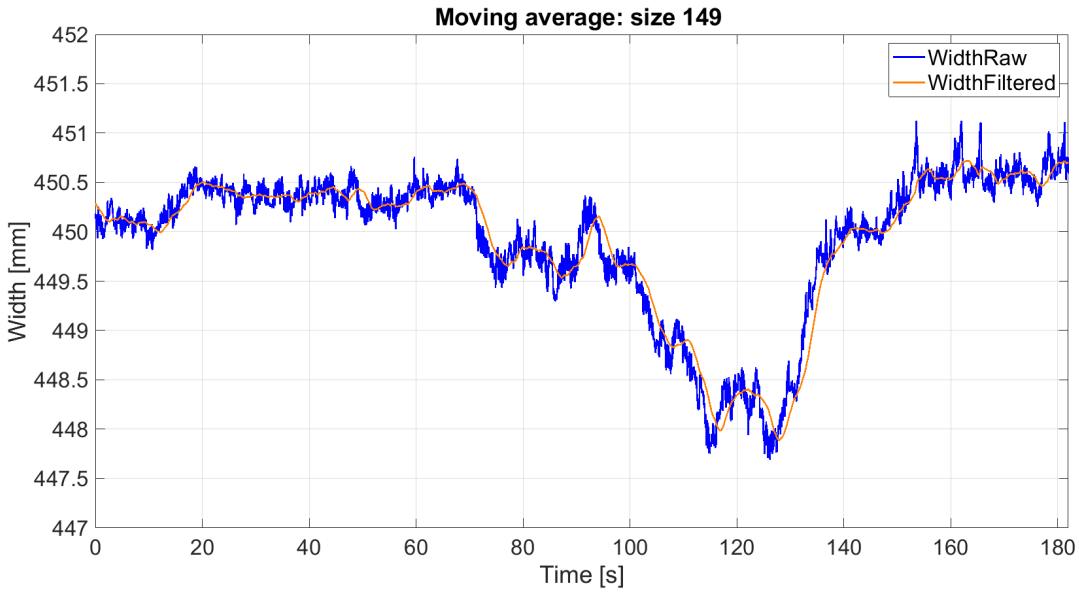

Figure 4.6: Width data with moving average filter applied (size 149)

To strike a balance between the limited smoothing of a small moving average and the high latency of a larger one, a first-order Bessel low-pass filter was employed. The Bessel filter provides an effective compromise by combining moderate filtering performance with a fast response time. Its principal advantage lies in its linear phase response. This minimises signal distortion and preserves the shape of transient signals

[53]. Although its attenuation beyond the cutoff frequency is less steep compared to filters such as Butterworth [54] or Chebyshev [55], the more gradual roll-off is acceptable in this context due to the improved time-domain behaviour and reduced risk of overshoot.

Figure 4.7 illustrates that the optimal cutoff frequency for this filter is approximately 0.06 Hz. At this setting, the filter is sufficiently aggressive to suppress uncorrectable high-frequency disturbances, while maintaining a modest time delay of around 0.25 seconds. This allows the control loop to remain responsive without overreacting to minor signal variations.

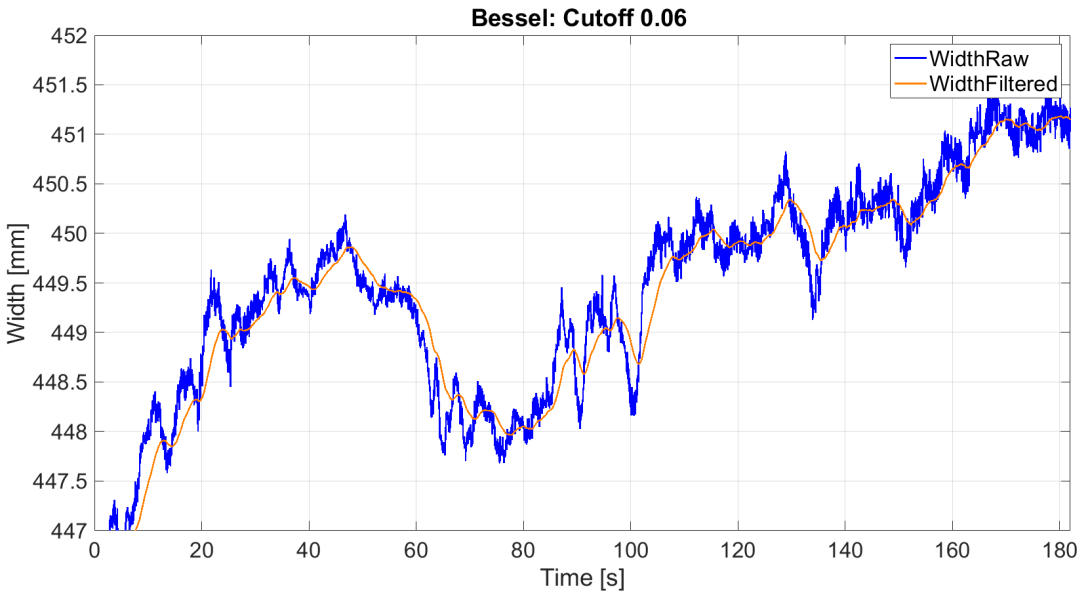

Figure 4.7: Width data with Bessel filter applied (Cutoff 0.06 Hz)

4.6 Tuning the proportional control gain

The final step in programming the tension control loop involves determining an appropriate gain value for the system. A common approach is to incrementally increase the gain on the machine until signs of instability begin to emerge, then reduce it to a stable level. However, given the availability of detailed measurement data collected in Chapter 3, this empirical tuning process may not be required. The existing data can instead support a more analytical estimation of the appropriate gain range, potentially reducing the reliance on extensive trial-and-error tuning. Moreover, this approach provides an opportunity for cross-validation between the measurement data presented in Chapter

3 and the control performance data from Chapter 4. Which can potentially enhance the overall consistency and robustness of the system design.

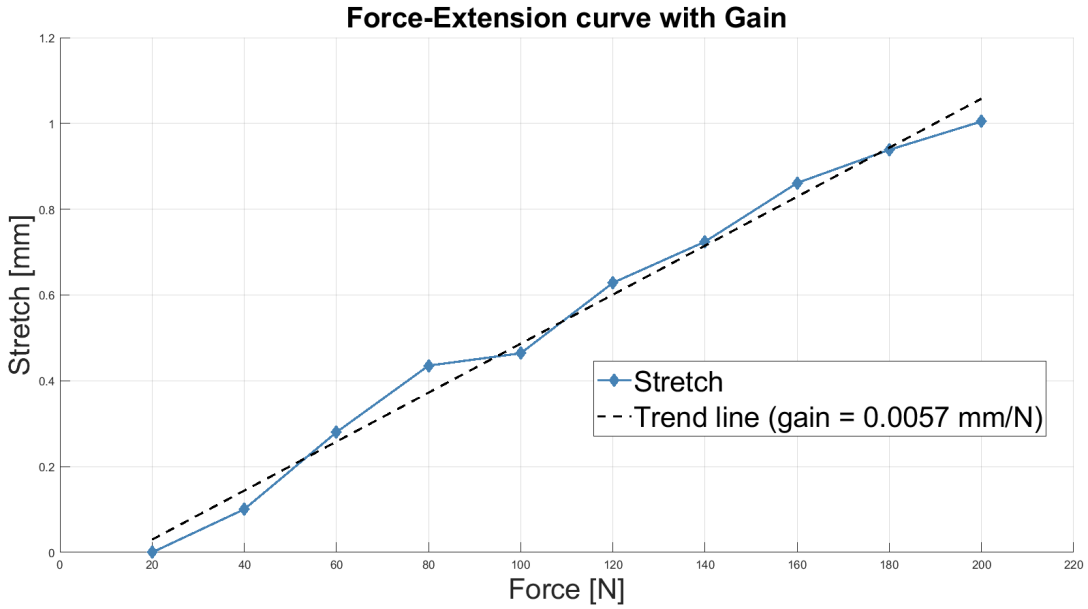

Figure 4.8: Force-extension Curve with trend line and Gain

Figure 4.8 presents the previously measured force–extension curve obtained using the RPi sensor. Since the tension control tests are conducted on the same roll of Kerdi, the material characteristics derived earlier offer a valuable reference for tuning the controller gain.

The gain extracted from this curve, expressed as stretch per unit of force, is approximately 0.0057 mm/N. However, the proportional controller requires the stiffness of the material, which is the inverse of this value, as given in Eq. 4.7:

$$\text{Stiffness} = \frac{1}{\text{Gain}} = \frac{1}{0.0057\,\text{mm/N}} \approx 175\,\text{N/mm} \tag{4.7}$$

It must also be taken into account that the material in question has undergone extensive use between the initial measurements and the testing of the control loop. To accommodate potential degradation, the gain for the P-controller was initially set to 100 based on the earlier data, and later adjusted to 80 in subsequent tests. This to reflect the reduced stiffness of the material over time. Larger gain values are likely feasible; however, the tension limits imposed by the dancer units rendered higher values impractical within

the current system configuration. The selected gain value is nonetheless more than sufficient for a proof-of-concept implementation and testing.

4.7 RPi sensor feedback-based quality control

Now that the control loop has been fully designed and implemented on the WPM using a combination of structured text and function blocks, the system can perform a comprehensive test. Figure 4.9 presents the results of one such test. The dark blue graph represents the material width measured while running the machine at a constant speed with a neutral applied tension of 20 N. This serves as a baseline reference for material quality in the absence of active control. The steel blue graph shows the material width measured during the activation of the tension control loop. The setpoint was deliberately chosen to coincide with the lowest point of the reference curve, in order to demonstrate the full operational range of the control system. From aggressive corrections where needed to minimal intervention when the material closely aligns with the setpoint. Both width graphs are plotted as a function of the material's position within the machine, as determined by the relative encoder of one of the winding rollers, rather than over time. This approach enables more effective visual comparison by aligning specific features of the material, thereby facilitating a more meaningful and qualitative inspection of the control loop's performance.

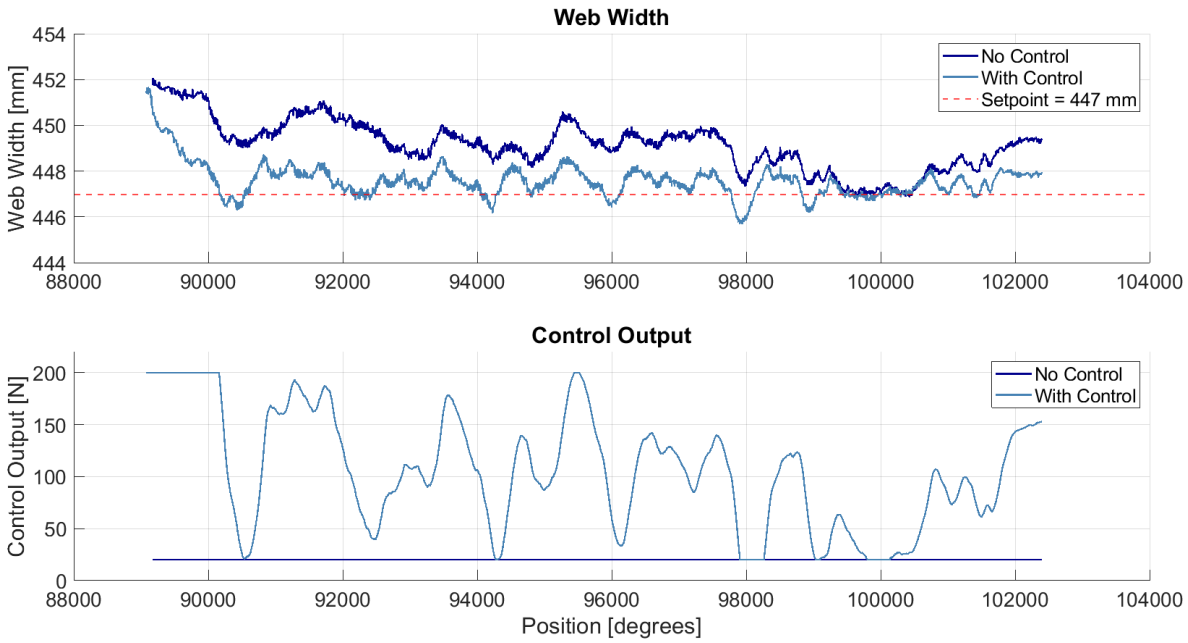

Figure 4.9: Width measurement before and during tension control and respective Dancer tension setpoint in function of material position

4.7.1 Metrics for evaluating material quality

To compare the two scenarios and quantify the effectiveness of the tension control in improving material quality, several performance metrics are employed. These metrics provide an objective basis for evaluating the extent to which the control loop enhances stability and reduces deformation.

Setpoint tracking and mean deviation

In order to quantify improvements in material straightness, the deviation of the web width from the target setpoint of 447 mm was evaluated before and after the control system was activated. While the absolute value of this setpoint is not intrinsically meaningful, it provides a consistent reference to assess how closely the control system is able to stabilise the width of the material.

The results in Table. 4.1 show a clear improvement in tracking performance. The mean width decreased from 449.17 mm in the uncontrolled case to 447.64 mm with control. This indicates that the average deviation from the setpoint was reduced by over 70%. Similarly, the Root Mean Square Error (RMSE) dropped from 2.44 mm to 1.01 mm, a reduction of approximately 59%. This demonstrates that the width of the material not

only shifts closer to the intended value, but also becomes more consistent and less prone to large deviations.

Importantly, while the absolute setpoint serves as a reference, these metrics primarily reflect the overall stability and reduction in large-scale variation. A lower RMSE implies fewer and smaller deviations from the mean path, which in practical terms equates to a straighter, more stable material flow.

Table 4.1: Setpoint Tracking Metrics

Metric	**Before Control**	**With Control**
Setpoint [mm]	447.00	447.00
Average width [mm]	449.17	447.64
Mean error [mm]	2.17	0.64
RMSE [mm]	2.44	1.01

Controllable width deviations (CDW)

While the average tracking error provides a useful indicator of width stability, it does not capture the dynamic shape of the material, which often includes large-scale warps or waves. These are better described by low-frequency fluctuations in the width signal, which are slow enough to be influenced by the control system. To isolate this controllable component, a low-pass filter with a cutoff frequency of 0.40 Hz was applied to the width signal, as seen in Fig. 4.10.

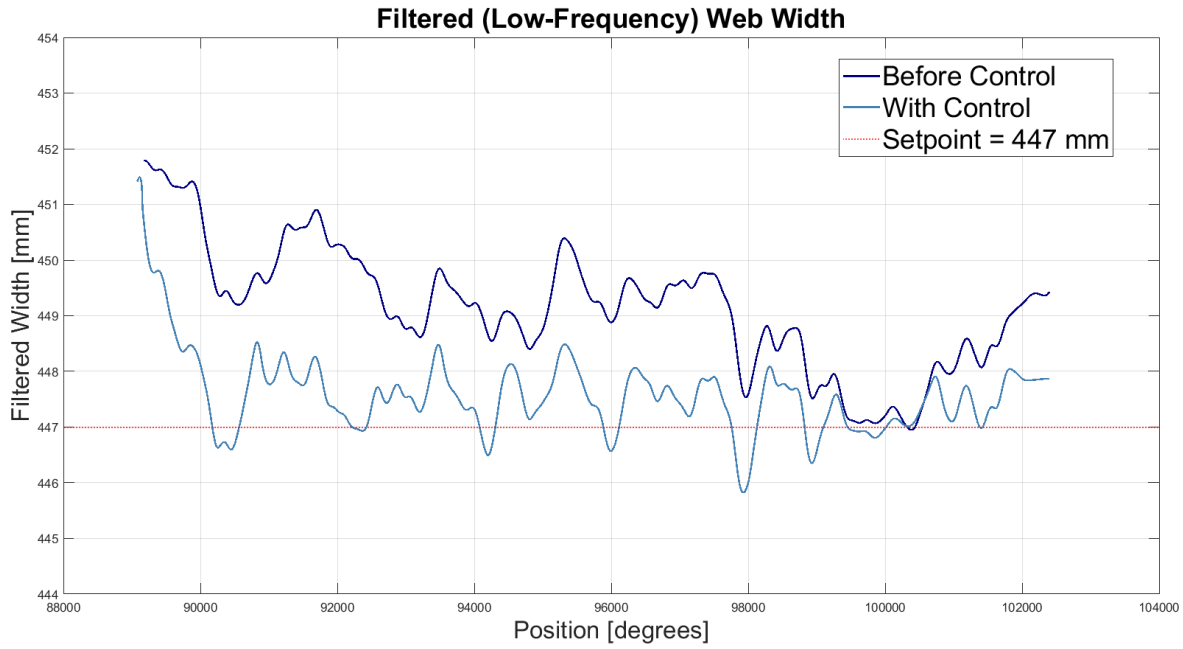

Figure 4.10: Measured material width using LP-filter cutoff 0.4 Hz

This cutoff was selected based on the estimated delay between actuation and sensing: after the material is tensioned by the dancer mechanism, it takes approximately 3.25 seconds before it reaches the width sensors. As a result, any fluctuations with a period shorter than this, corresponding to frequencies above roughly 0.30–0.35 Hz, cannot realistically be corrected in time by the controller. Setting the cutoff at 0.40 Hz ensures that all frequency components the controller could reasonably affect are included, while higher-frequency components (e.g., wrinkles, noise) are excluded and treated separately.

Table 4.2 summarises two metrics derived from the filtered width signal: the standard deviation, which reflects the general variability of low-frequency warps, and the range, which measures the peak-to-peak spread. Without control, the SD of this low-frequency component was 1.10 mm, compared to 0.76 mm after activation, a reduction of approximately 30%. This indicates a substantial improvement in straightness by reducing slow, large-amplitude deviations in the material.

The range, however, increased slightly from 4.86 mm to 5.66 mm. This is likely due to a pronounced initial correction at the start of the controlled sequence, where the dancer rapidly pulls the material toward the setpoint. While this causes a larger initial deviation, it does not negate the observed reduction in long-term width variability.

Finally, the high-frequency residual, representing noise and fast fluctuations outside the controller's influence, remained effectively unchanged (standard deviation: 0.106 mm vs 0.109 mm), as seen in Fig. 4.11. This confirms that the control action is correctly limited to the slow components and does not introduce unintended oscillations.

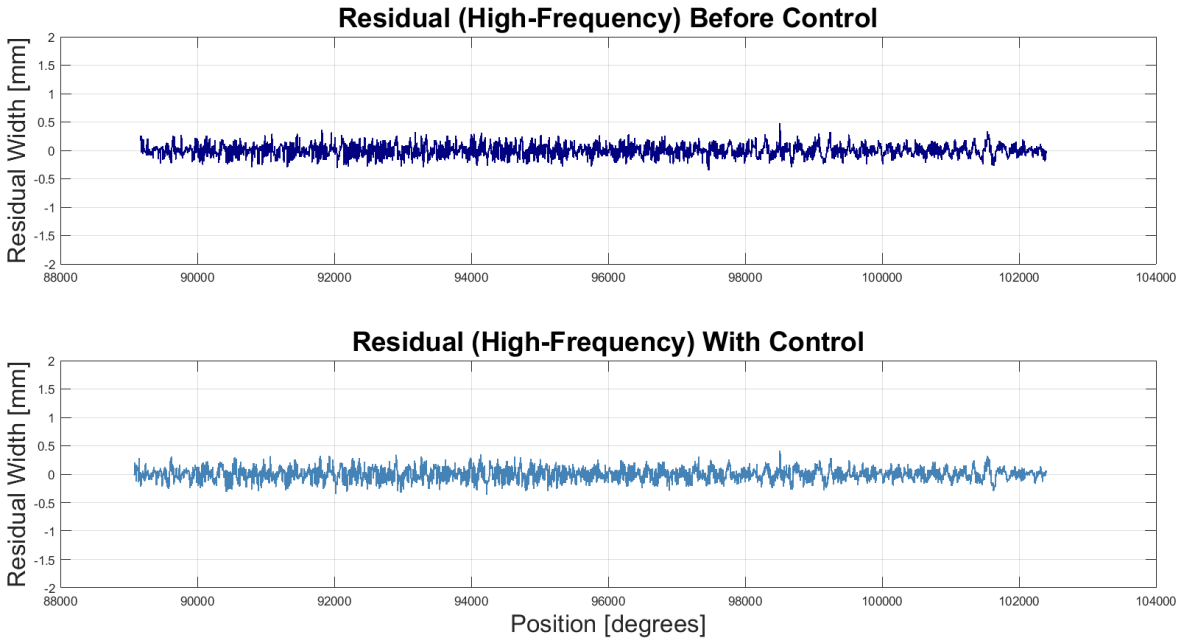

Figure 4.11: Material width residual high frequency noise

Table 4.2: Controllable Width Deviations Metrics

Metric	**Before Control**	**With Control**
CWD SD [mm]	1.10	0.76
CWD range [mm]	4.86	5.66
High-frequency residual SD [mm]	0.106	0.109

Frequency-domain analysis

To further understand how the control system affects the structure of the material, the width signal was analysed in the frequency domain. By applying a Fourier transform, the total width variation was decomposed into different frequency components, corresponding to slow and fast deformations, the results of which can be seen in Fig. 4.12.

The spectral energy was divided into three bands:

- Low-frequency components (below 0.20 Hz), which represent large, slow warps in the material.
- Control-band components (0.20 Hz to 0.40 Hz), near the upper limit of the controller's responsiveness.
- High-frequency components (above 0.40 Hz), which correspond to fast fluctuations or fine wrinkles that the control system cannot act upon.

The spectral energy in the low-frequency band was reduced from 0.595 (arbitrary units (a.u.)) to 0.274, indicating that the control system effectively suppressed large-scale deformations. This aligns with earlier findings based on time-domain filtering. The control-band energy increased from 0.0067 to 0.0158, which can be attributed to two complementary effects. First, the controller operating near its bandwidth limit may introduce small overshoots or ripple, slightly amplifying components within this range. Second, as the controller responds to slow deformations, it does so with finite reaction time. This means that part of the energy originally present in the low-frequency band is effectively redistributed into the control band. In other words, low-frequency deviations are not instantaneously suppressed but instead transformed into compensatory responses that manifest at slightly higher frequencies. High-frequency energy increased marginally as well (0.0091 to 0.0126), which is expected. The controller does not actively suppress these components and may indirectly introduce some high-frequency effects due to actuation dynamics.

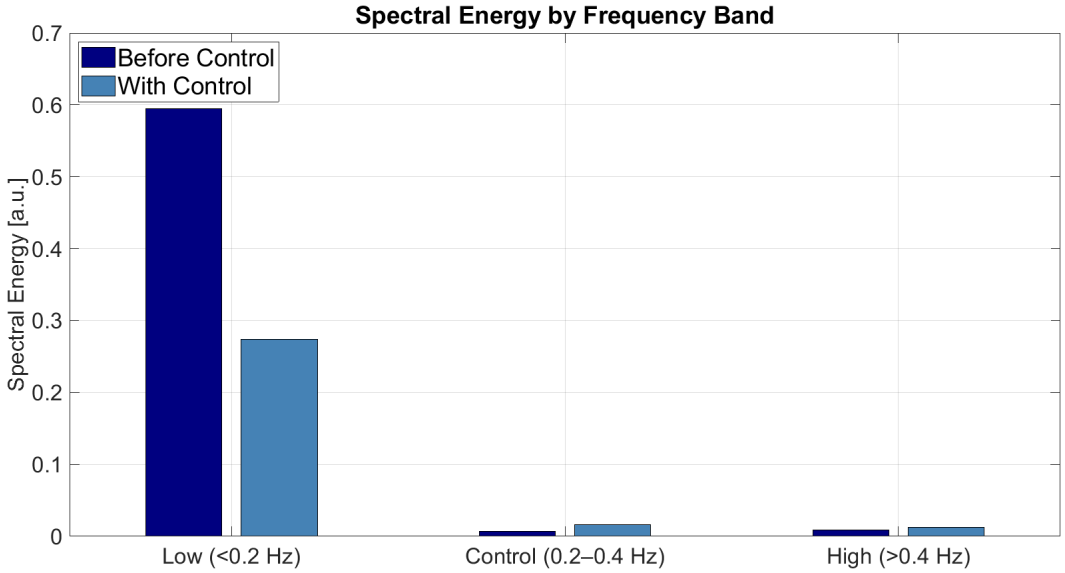

Figure 4.12: Spectral Energy by Frequency Band

To summarise these changes into a single indicator of straightness, the total spectral energy across all bands was computed. Since all frequency ranges were given equal importance (weights of 1), the resulting measure reflects the total deformation energy in the frequency domain. This value decreased from 0.611 to 0.302, a 50% reduction, confirming that the controller effectively flattens the material by eliminating low-frequency deviations without introducing substantial noise elsewhere.

Table 4.3: Frequency-Domain Analysis Metrics

Frequency Band	**Before Control**	**With Control**
Low frequency < 0.20 Hz [a.u.]	0.595	0.274
Control band 0.20–0.40 Hz [a.u.]	0.0067	0.0158
High frequency > 0.40 Hz [a.u.]	0.0091	0.0126
Total spectral energy [a.u.]	0.611	0.302

4.8 Nuanced interpretation of control behaviour and test setup

It is important to note that material quality in this study was evaluated based on straightness, assuming that a perfectly linear width profile corresponds to ideal material behaviour. While this is a useful proxy for detecting warps and irregularities, it does not fully capture the complexity of quality assessment. In practice, aggressively tensioning the material to straighten it can introduce plastic deformation or weaken the structure, effectively reducing the material's functional quality. The real objective of the proposed control system is therefore not to force the material into a straight shape, but to dynamically adjust the applied tension in response to machine-induced deformations. By compensating for variations introduced by components such as misaligned rollers or uneven traction, the system aims to maintain the natural form of the material as it would exist under neutral tension.

One such source of periodic deformation within the WPM is the pivot table, a component known to introduce characteristic stretch patterns into the material during normal operation, as can be seen in Fig. 4.13. In principle, these deformations are precisely the kind that the proposed control loop is designed to detect and compensate for in real time. Unfortunately, the pivot table was not functioning during the limited measurement campaign window of this thesis, which prevented a direct demonstration of compensation

for these real-world disturbances. As a result, the test sequence was performed under deliberately aggressive conditions, using a width setpoint chosen to provoke strong controller responses. These corrections should therefore be interpreted primarily as a proof-of-concept of the sensor and control strategy's responsiveness, rather than an optimal or quality-preserving implementation.

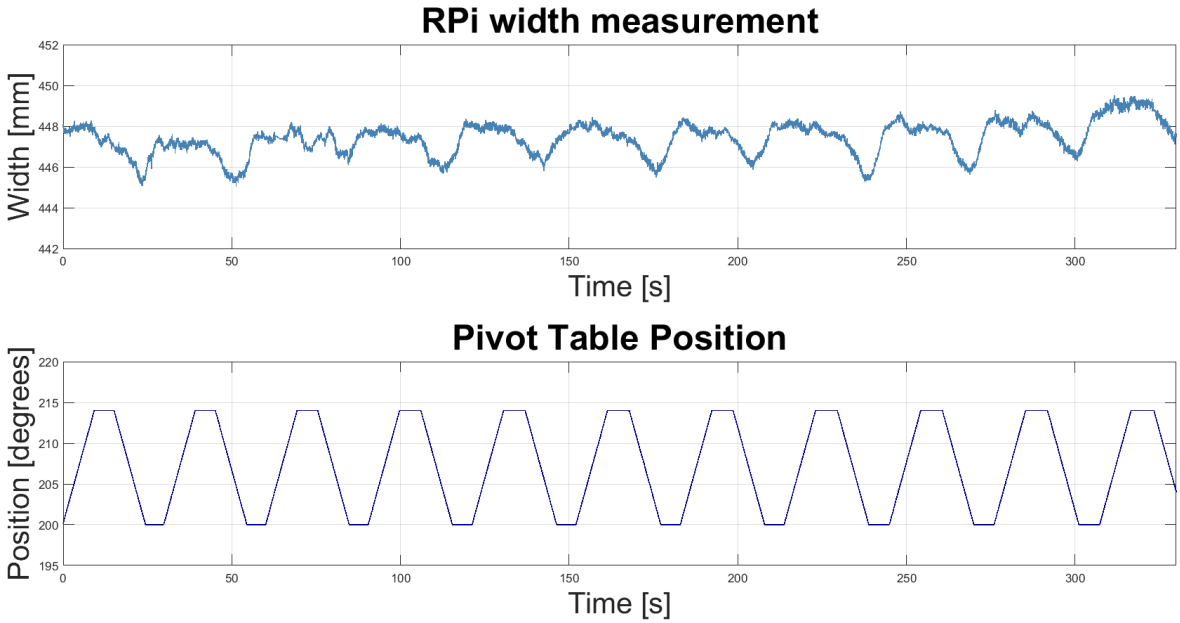

Figure 4.13: Correlation between material deformation and pivot table position showing pivot table influence on material width

4.9 Summary of use case phase

This chapter presents the design and implementation of a closed-loop tension control system for the WPM, which uses real-time feedback from a RPi-based stretch sensor. A proportional controller is added upstream of the existing dancer PI controller, allowing the applied tension to adjust dynamically in response to measured width deviations. The objective is to reduce large-scale material deformations and improve overall straightness during processing.

The RPi sensor demonstrates sufficient precision and sampling frequency to detect relevant fluctuations in material width. However, the three-second delay between actuation at the dancer and sensing at the measurement location imposes a fundamental limitation. This delay restricts the control loop's effective range to relatively slow, low-

frequency deformations. Time-domain metrics and frequency-domain analysis both reflect this bandwidth limitation.

Despite these constraints, the results suggest a meaningful improvement in material behaviour under control. Reductions in controllable width deviations and low-frequency spectral energy of up to 70% indicate a tangible enhancement in material behaviour under control. While these reductions are context-dependent and not conclusive, they indicate that the system is capable of mitigating low-frequency warps and improving stability in a measurable way.

The proportional gain is estimated based on the force–extension curve derived earlier in Chapter 3. A stiffness value of approximately 175 N/mm leads to an initial controller gain of 100, which is later adjusted to 80 to account for observed material degradation. Although more advanced tuning may improve performance further, the chosen values are sufficient for a functional proof-of-concept within the mechanical constraints of the current system.

Overall, while this control approach does not address all possible material deformations, it demonstrates that the RPi sensor is capable of delivering sufficiently precise, timely, and consistent measurements to support real-time feedback in an industrial context. The integration of this low-cost, modular sensor into a functional control loop highlights its potential as a viable component in future closed-loop quality control systems. Its ability to detect and quantify slow, large-scale deformations with high spatial resolution makes it particularly well-suited for applications where traditional high-end sensors may be impractical or cost-prohibitive. As such, this work underscores the feasibility of using embedded, accessible sensor platforms to enable responsive, data-driven control in web processing environments.

Conclusion

This thesis set out to explore the feasibility and performance of a low-cost, vision-based stretch sensor for web processing machines. This with the broader aim of contributing to smarter and more accessible quality control solutions within the context of Industry 4.0. By integrating embedded vision, lightweight edge detection algorithms, and real-time feedback mechanisms, the work sought to demonstrate that affordable hardware platforms could meaningfully address a class of industrial measurement challenges typically dominated by high-end metrology systems.

Throughout the thesis, a number of specific research questions and technical goals were articulated. These included the development of a robust and autonomous sensor architecture, the achievement of precision within a targeted range (20–100 µm), and the validation of the sensor's industrial viability through comparison with a commercial ultrasonic sensor. In addition, the work aimed to explore how the proposed system might be integrated into closed-loop control systems to actively enhance material quality within a WPM environment.

The development process revealed both the promise and limitations of embedded vision-based sensing. Through careful algorithmic design, most notably the application of Frequency Domain Zero Padding and parallel processing techniques, subpixel edge detection was achieved. The estimated measurement precision ranged between 2.8 and 28.3 µm, depending on material characteristics and camera configuration. These results not only exceeded the initial precision targets but also compared favourably against the industrial ultrasonic edge sensor. This was particularly evident in scenarios involving lightweight or semi-transparent materials, where conventional sensors often struggle.

However, these precision gains came at the cost of reduced cycle frequency. The achieved measurement rate of approximately 30 Hz fell short of the targeted 50–500 Hz range. This highlights a fundamental performance trade-off inherent in the use of affordable, general-purpose computing platforms such as the Raspberry Pi. While this limitation does constrain the system's applicability to high-speed processes, it may still prove acceptable in many real-world use cases where material dynamics evolve more slowly.

Subsequent evaluation under dynamic conditions further substantiated the practical utility of the system. In comparative experiments on a functioning WPM, the RPi sensor was shown to reliably track material deformation trends with high fidelity and signal

clarity. Measurements of stretch correlated closely with those of the ultrasonic sensor, and the system demonstrated sufficient sensitivity to distinguish both elastic and plastic deformation behaviours in the tested materials. These outcomes were encouraging and suggested that the sensor could indeed be integrated into feedback loops for online quality control.

This possibility was further investigated through the design and implementation of a closed-loop tension control scheme that used sensor feedback to regulate material stretch. Although the system's effective control bandwidth was ultimately limited by a three-second delay between actuation and sensing, the controller nonetheless achieved meaningful improvements in material straightness. Both time-domain and frequency-domain metrics, including reductions in low-frequency spectral energy and standard deviation, supported the conclusion that the system mitigated large-scale material warps under controlled operation.

Taken together, these findings offer a promising, albeit preliminary case for the deployment of low-cost vision-based sensors in industrial contexts where cost, adaptability, and integration ease are key priorities. It would be premature to suggest that such systems can supplant high-end alternatives across the board; rather, the evidence points toward a valuable complementary role, especially in applications involving atypical materials, constrained budgets, or legacy equipment without existing sensing infrastructure.

There remain several avenues for future work. The current sensor architecture may benefit from hardware acceleration, for instance through the use of more powerful microcontrollers, which could address the cycle time bottleneck. Further, the integration of multiple sensing modalities, such as combining vision with force or displacement sensors, could yield more comprehensive assessments of material behaviour. Finally, broader validation across a wider range of materials and operating conditions would help to more rigorously define the system's limitations and potential.

4 Conclusion

In summary, this thesis demonstrates that vision-based, low-cost stretch sensing is technically viable and may hold considerable promise for enhancing the accessibility and versatility of quality control in web processing machines. Although challenges remain, particularly regarding speed and scalability, the presented system constitutes a solid proof-of-concept and a compelling contribution to the growing field of affordable industrial sensing.

Synopsis

Key achievements of this thesis include:

- **Precision:** The system achieved a measurement precision between 2.8 and 28.3 µm, outperforming a commercial ultrasonic sensor in certain cases.
- **Validation:** Comparison with ultrasonic sensors under dynamic conditions demonstrated strong signal correlation and improved noise performance.
- **Integration:** A closed-loop control system using the sensor's output successfully reduced low-frequency material deformation by 50–70%.
- **Industrial Potential:** The sensor was shown to be cost-effective, adaptable, and suitable for scenarios involving difficult-to-measure materials such as semi-transparent meshes.

Future Work

This thesis shows that a low-cost vision system can precisely measure material stretch inside a WPM. The system works well, but several improvements are possible. This section presents some ideas for future work, based on the main chapters of the thesis.

Leveraging Multithreading for Higher Resolution

The original system processed images sequentially: it captured a frame, then processed it, with each step blocking the other. This limited performance and left CPU and camera time underused. A multithreaded architecture now allows image capture and edge detection to run in parallel. This significantly improves throughput and opens the door to higher-resolution processing without increasing cycle time. Higher resolution improves accuracy and reduces the required FDZP factor. With the freed-up processing time, it may also be possible to slightly increase the FDZP Factor again, improving subpixel precision. Future work should explore the optimal trade-off between resolution, FDZP scaling, and cycle time under this new structure.

Hardware Optimisation and Platform Migration

The RPi version 4B offers a good balance between cost and performance. However, it has limitations. It throttles under heat, lacks a GPU, and has no real-time clock. Upgrading to an RPi version 5 or using a more powerful board like the NVIDIA Jetson Nano can help. These platforms support faster processing and better multithreading. They can also run more complex image algorithms. Stereo vision or lightweight neural networks may then become possible.

Synchronisation and Time-Stamping

Each RPi works independently in the current setup. This causes small timing errors between the two measurements. Future systems should synchronise the cameras more precisely. GPIO triggers or real-time clocks can achieve this. Another option is to add timestamps to each frame and align them in software. These changes would reduce width calculation errors, especially at high material speeds.

4 Future Work

Validating Control Loop with Pivot Table Influence

The next major step is to test the sensor-integrated control loop on a WPM with an active pivot table. Previous tests used a simplified setup without pivot-induced deformation. By comparing runs with and without pivot movement, it becomes possible to evaluate whether the current control loop can compensate for these added disturbances. This will show how well the system maintains stable material width under realistic conditions. The ultimate goal of this validation is to ensure consistent material quality. If the control loop can successfully handle deformations caused by pivot table motion, it proves the system's ability to maintain dimensional stability during production. Such control is essential for achieving reliable, high-quality output in web-based manufacturing.

Building a Digital Twin for Model-Based Compensation

In parallel, future work should explore how pivot table movements affect the material in detail. By collecting data on table angle, position, and speed alongside stretch sensor measurements, one can analyse how different motions influence material stress and deformation. With this data, it may be possible to build a simplified digital twin of the pivot table. Such a model could estimate internal stresses and predict resulting width changes based solely on table movement. In the long term, this model could replace the sensor feedback in the control loop, enabling predictive, feedforward compensation for pivot-induced deformation, moving the system closer to a fully virtualised, smart manufacturing environment.

Societal Reflection

The development of a low-cost, vision-based stretch sensor for WPMs contributes to the ongoing transformation of industrial production systems. This work aligns with broader societal shifts such as the digitisation of manufacturing, the rise of smart factories, and the increasing push for sustainability and transparency in production processes.

Broader Societal Context

Today's manufacturing landscape is undergoing rapid digital transformation. Smart sensors and data-driven systems form the foundation of Industry 4.0. By introducing an affordable, embedded computer vision solution for material monitoring, this thesis directly contributes to making intelligent manufacturing more accessible, not only to large corporations but also to smaller production lines with limited budgets. Enabling such access helps democratise innovation and supports economic resilience in smaller or less-automated sectors.

In this sense, the developed sensor system reflects a key societal trend: the growing demand for cost-effective, modular technologies that increase control and traceability without the need for expensive infrastructure upgrades. It supports a vision of industry where digital tools are integrated at every level, from operator feedback to closed-loop control systems.

Relevance to the Sustainable Development Goals (SDGs)

This work supports several of the United Nations SDGs. Most notably:

- **Goal 9:** Industry, Innovation and Infrastructure: By designing a low-cost, replicable system that improves quality control in industrial settings, the project promotes inclusive and sustainable industrialisation and fosters innovation.
- **Goal 12:** Responsible Consumption and Production: Improved material monitoring allows manufacturers to detect stretch and deformation early, which can reduce material waste and energy use. This contributes directly to more sustainable production processes.
- **Goal 8:** Decent Work and Economic Growth: Empowering smaller manufacturers with intelligent tools can strengthen their competitiveness, improve efficiency, and create opportunities for skilled technical jobs in process monitoring and automation.

4 Societal Reflection

Lifecycle and System-Level Sustainability

In addition to reducing immediate waste and improving production efficiency, the proposed system also contributes to sustainability on a systems level. Because it relies on affordable, off-the-shelf components and open software tools, the sensor is easy to replicate, repair, and upgrade. This extends its usable lifetime and reducing reliance on proprietary industrial systems that often require full replacement when obsolete. This modularity supports circular design principles and aligns with the growing need for adaptable, low-footprint solutions in a rapidly evolving manufacturing landscape. As industries seek to transition to more flexible, resource-conscious production, such scalable tools can play a crucial role in reducing environmental impact across the entire lifecycle of manufacturing equipment.

References

[1] N. Jazdi, “Cyber physical systems in the context of industry 4.0,” in *2014 IEEE International Conference on Automation, Quality and Testing, Robotics*. IEEE, 2014, pp. 1–4.

[2] A. Papavasileiou, G. Michalos, and S. Makris, “Quality control in manufacturing – review and challenges on robotic applications,” *International Journal of Computer Integrated Manufacturing*, vol. 38, no. 1, pp. 79–115, 2025. [Online]. Available: https://doi.org/10.1080/0951192X.2024.2314789

[3] H. Kagermann, W. Wahlster, and J. Helbig, “Securing the future of german manufacturing industry: Recommendations for implementing the strategic initiative industrie 4.0,” acatech – National Academy of Science and Engineering, Frankfurt/Main, Germany, Tech. Rep., April 2013, final report of the Industrie 4.0 Working Group. [Online]. Available: https://www.acatech.de/publikation/securing-the-future-of-german-manufacturing-industry/

[4] eFunda, “Strain gage sensitivity,” n.d. [Online]. Available: https://www.efunda.com/designstandards/sensors/strain_gages/strain_gage_sensitivity.cfm

[5] Unknown, “Strain gauge measurement: A tutorial,” n.d. [Online]. Available: https://web.enib.fr/~bourgeot/IPS/Datasheets/AN078-Strain_Gauge_Measurement-A_Tutorial.pdf

[6] Y. W. Teyeme, B. Malengier, T. Tesfaye, and L. Van Langenhove, “A fabric-based textile stretch sensor for optimized measurement of strain in clothing,” *Sensors*, vol. 2020, 2020, pug01:8685720. [Online]. Available: http://lib.ugent.be/catalog/pug01:8685720

[7] National Instruments, “Measuring strain with strain gages,” 2025. [Online]. Available: https://www.ni.com/en/shop/data-acquisition/sensor-fundamentals/measuring-strain-with-strain-gages.html

[8] P. H. Mott and C. M. Roland, “Limits to poisson’s ratio in isotropic materials,” *Physical Review B*, vol. 80, no. 13, p. 132104, 2009. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevB.80.132104

[9] H. Xiufeng, W. Jihang, and Z. Zhenhua, “Research on application of digital image correlation method,” in *2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, 2020, pp. 838–840.

4 References

[10] J. Carr, J. Baqersad, C. Niezrecki, and P. Avitabile, “Predicting dynamic strain on wind turbine blade using digital image correlation techniques in conjunction with analytical expansion methodologies,” *Proceedings of the 31st International Modal Analysis Conference (IMAC)*, 2013. [Online]. Available: https://www.researchgate.net/publication/292783598_Predicting_Dynamic_Strain_on_Wind_Turbine_Blade_Using_Digital_Image_Correlation_Techniques_in_Conjunction_with_Analytical_Expansion_Methodologies

[11] R. E, A. A, V. V, G. T, S. S, and S. Prabha R, “Strain measurement using digital image correlation technique,” in *2023 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, 2023, pp. 1–8.

[12] AIMed Research, “High-speed cameras,” n.d. [Online]. Available: https://www.aimed-research.com/high-speed-cameras.html

[13] J. Bruggeman, “Operational motion monitoring of web processing machines using low-cost vision for adaptive control,” 2024, master of Science in Electromechanical Engineering. [Online]. Available: http://lib.ugent.be/catalog/rug01:003214547

[14] D. Marr and E. Hildreth, “Theory of edge detection,” *Proceedings of the Royal Society of London. Series B, Biological Sciences*, vol. 207, pp. 187–217, 1980. [Online]. Available: https://doi.org/10.1098/rspb.1980.0020

[15] L. Ding and A. Goshtasby, “On the canny edge detector,” *Pattern Recognition*, vol. 34, no. 3, pp. 721–725, 2001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320300000236

[16] M. S. I. Khan, “Implementation of edge & shape detection techniques and their performance evaluation,” Master’s thesis, Ryerson University, Toronto, Ontario, Canada, 2013, available at Digital Commons @ Ryerson. [Online]. Available: http://digitalcommons.ryerson.ca/dissertations/1043

[17] A. unknown, “A comparison of various edge detection techniques used in image processing,” *ResearchGate*, 2016. [Online]. Available: https://www.researchgate.net/publication/303142762_A_Comparison_of_various_Edge_Detection_Techniques_used_in_Image_Processing

[18] Florida State University, “Digital image processing - difference of gaussians,” n.d. [Online]. Available: https://micro.magnet.fsu.edu/primer/java/digitalimaging/processing/diffgaussians/index.html

4 References

[19] G. Verhoeven, W. Karel, S. Štuhec, M. Doneus, I. Trinks, and N. Pfeifer, "Mind your grey tones: Examining the influence of decolourization methods on interest point extraction and matching for architectural image-based modelling," in *Proceedings of the International Conference on Computer Vision and Pattern Recognition in Remote Sensing*, D. González-Aguilera, F. Remondino, J. Boehm, T. Kersten, and T. Fuse, Eds. Göttingen, 2015. [Online]. Available: http://lib.ugent.be/catalog/pug01:5857128

[20] International Telecommunication Union, "Recommendation ITU-R BT.601-7: Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios," International Telecommunication Union, Tech. Rep. BT.601-7, 2011. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/bt/r-rec-bt.601-7-201103-i!!pdf-e.pdf

[21] Y. Thielemans, N. Kayedpour, A. Coene, G. Crevecoeur, and J. D. M. De Kooning, "Speed and position measurement of rotating machinery with a triangular pattern vision encoder," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1–13, 2025.

[22] M. Heideman, D. Johnson, and C. Burrus, "Gauss and the history of the fast fourier transform," *IEEE ASSP Magazine*, vol. 1, no. 4, pp. 14–21, 1984.

[23] L. Auslander and F. A. Grunbaum, "The fourier transform and the discrete fourier transform," *Inverse Problems*, vol. 5, pp. 149–164, 1989. [Online]. Available: https://api.semanticscholar.org/CorpusID:121312591

[24] J. Wolf, "On the classification of hermitian symmetric spaces," *Indiana University Mathematics Journal*, vol. 13, pp. 489–495, 1964. [Online]. Available: https://www.iumj.indiana.edu/IUMJ/fulltext.php?artid=13041

[25] S. E. Kelly, "Gibbs phenomenon for wavelets," *Applied and Computational Harmonic Analysis*, vol. 3, no. 1, pp. 72–81, 1996. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1063520396900068

[26] C. R. Taylor, *Applications of Dynamic Programming to Agricultural Decision Problems*. CRC Press, 2017. [Online]. Available: https://books.google.be/books?id=71SsDwAAQBAJ

[27] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

4 References

[28] A. Bousaid, T. Theodoridis, S. Nefti-Meziani, and S. Davis, “Perspective distortion modeling for image measurements,” *IEEE Access*, vol. 8, pp. 15 322–15 331, 2020.

[29] J. Watt. (2024) How to fast remove fisheye (lens distortion) from gopro video. [Online]. Available: https://www.winxdvd.com/resource/gopro-fisheye-correction-guide.htm

[30] J. Jedlička and M. Potůčková, “Correction of radial distortion in digital images,” *ResearchGate*, 2007, accessed: 2025-03-12. [Online]. Available: https://www.researchgate.net/publication/255610429_CORRECTION_OF_RADIAL_DISTORTION_IN_DIGITAL_IMAGES

[31] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[32] O. Team, “Camera calibration with opencv,” 2024. [Online]. Available: https://docs.opencv.org/4.x/dc/dbb/tutorial_py_calibration.html

[33] Python Software Foundation, “multiprocessing — process-based parallelism,” n.d. [Online]. Available: https://docs.python.org/3/library/multiprocessing.html

[34] Real Python, “An intro to threading, multiprocessing, and asyncio in python,” n.d. [Online]. Available: https://realpython.com/python-concurrency/

[35] ARM Developer, “Multicore processing,” n.d. [Online]. Available: https://developer.arm.com/documentation/102984/0100/Multicore-Processing

[36] F. Doglio, *Parallel Programming with Python*, 2nd ed. Packt Publishing, 2014. [Online]. Available: https://www.packtpub.com/product/parallel-programming-with-python-9781783288397

[37] Commsvr, “Opc unified architecture,” n.d. [Online]. Available: https://commsvr.gitbook.io/ooi/introduction/opcunifiedarchitecture

[38] Python OPC UA, “Python opc ua library documentation,” n.d. [Online]. Available: https://python-opcua.readthedocs.io/en/latest/

[39] GeeksforGeeks, “Tcp/ip model,” n.d. [Online]. Available: https://www.geeksforgeeks.org/tcp-ip-model/

[40] Siemens AG, “Opc ua methods for the simatic s7-1500 opc ua server,” n.d. [On-

line]. Available: https://support.industry.siemens.com/cs/document/109756885/opc-ua-methods-for-the-simatic-s7-1500-opc-ua-server-

[41] AutonomyLogic, *OpenPLC Runtime Documentation*, 2022. [Online]. Available: https://autonomylogic.com/docs-category/openplc-runtime/

[42] *Accuracy (trueness and precision) of measurement methods and results — Part 1: General principles and definitions*, International Organization for Standardization Std. ISO 5725-1:2023, July 2023. [Online]. Available: https://cdn.standards.iteh.ai/samples/69418/a7c3fb78cdce4a6398bd3f759670d72e/ISO-5725-1-2023.pdf

[43] Raspberry Pi Foundation. (2019) Thermal testing raspberry pi 4. [Online]. Available: https://www.raspberrypi.com/news/thermal-testing-raspberry-pi-4/

[44] J. Geerling. (2020) The raspberry pi 4 has a fan now – the case fan. [Online]. Available: https://www.jeffgeerling.com/blog/2020/raspberry-pi-4-has-fan-now-case-fan

[45] M. GmbH, “bks+6/fiu ultrasonic web edge sensor,” https://sensorpartners.com/en/product/microsonic-bks6-fiu/, 2025.

[46] Study.com. (2024) How to calculate the standard deviation of the sum of two random variables. [Online]. Available: https://study.com/skill/learn/how-to-calculate-the-standard-deviation-of-the-sum-of-two-random-variables-explanation.html

[47] Statistics How To. (2024) Correlation coefficient: Simple definition, formula, easy steps. [Online]. Available: https://www.statisticshowto.com/probability-and-statistics/correlation-coefficient-formula/

[48] Kent State University Libraries. (2024) Spss tutorials: Pearson correlation. [Online]. Available: https://libguides.library.kent.edu/SPSS/PearsonCorr

[49] Schluter Systems, “Schluter®-KERDI: Waterproofing Membrane,” https://www.schluter.com/schluter-us/en_US/Membranes/Waterproofing-%28KERDI%29/Schluter%C2%AE-KERDI/p/KERDI, 2025.

[50] (2024) Polyethylene (pe) | properties, structures, uses, & facts. Encyclopædia Britannica. [Online]. Available: https://www.britannica.com/science/polyethylene

[51] (2024) Polypropylene | properties, definition, & uses. Encyclopædia Britannica. [Online]. Available: https://www.britannica.com/science/polypropylene

[52] (2025) Capstan equation - an overview | sciencedirect topics. ScienceDirect.

[Online]. Available: https://www.sciencedirect.com/topics/engineering/capstan-equation

[53] (2025) Bessel filter - an overview | sciencedirect topics. ScienceDirect. [Online]. Available: https://www.sciencedirect.com/topics/engineering/bessel-filter

[54] (2025) Butterworth filter - an overview | sciencedirect topics. ScienceDirect. [Online]. Available: https://www.sciencedirect.com/topics/engineering/butterworth-filter

[55] (2025) Chebyshev filter - an overview | sciencedirect topics. ScienceDirect. [Online]. Available: https://www.sciencedirect.com/topics/engineering/chebyshev-filter

Appendices

Appendix A

Camera Matrix and Distortion Coefficients

The intrinsic parameters of the pinhole camera model are represented by the **camera matrix**:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Where:

- f_x, f_y: focal lengths in pixel units
- c_x, c_y: coordinates of the principal point (usually near the image centre)

The **distortion vector** models radial and tangential lens distortion:

$$\mathbf{d} = \begin{bmatrix} k_1 & k_2 & p_1 & p_2 & k_3 \end{bmatrix}$$

Distorted image coordinates (x_d, y_d) are computed from undistorted normalized coordinates (x, y) as:

$$r^2 = x^2 + y^2$$

$$x_d = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 xy + p_2(r^2 + 2x^2)$$

$$y_d = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2y^2) + 2p_2 xy$$

These parameters are estimated during intrinsic calibration using known patterns (e.g., a ChArUco board), enabling undistortion and accurate metric measurements.